

AKD PDMM®

User Guide



Edition September 2016, Revision G

Valid for Hardware Revision EB

Valid for Firmware Revision 1.15

Part Number 903-200016-00



Keep all manuals as a product component during the life span of the product.
Pass all manuals to future users/owners of the product.

KOLLMORGEN®

Because Motion Matters™

Hardware Revision (HR)

| AKD-B/P-NA | AKD-B/P-NB | AKD-M | AKD-T-IC | Firmware/Workbench | KAS IDE | Export Classification | Remarks |
|------------|------------|-------|----------|--------------------|----------|-----------------------|--|
| A | - | - | - | from 1.3 | - | 3A225 | Start revision, export control |
| C | - | - | - | from 1.5 | - | 3A225 | STO certified, PROFINET RT released, export control |
| - | D | DB | DA | from 1.6 | from 2.5 | 3A225 | Control board revision 9, AKD PDMM Start revision, AKD BASIC-IC Start revision, export control |
| D | E | EB | EA | from 1.13 | from 2.9 | - | Hardware revision for export classification traceability purposes |

Trademarks

- AKD is a registered trademark of Kollmorgen Corporation
- SynqNet is a registered trademark of Motion Engineering Inc.
- EnDat is a registered trademark of Dr. Johannes Heidenhain GmbH
- EtherCAT is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH
- Ethernet/IP is a registered trademark of ODVA, Inc.
- Ethernet/IP Communication Stack: copyright (c) 2009, Rockwell Automation
- sercos® is a registered trademark of sercos® international e.V.
- HIPERFACE is a registered trademark of Max Stegmann GmbH
- PROFINET is a registered trademark of PROFIBUS and PROFINET International (PI)
- SIMATIC is a registered trademark of SIEMENS AG
- Windows is a registered trademark of Microsoft Corporation

Current patents

- US Patent 5,162,798 (used in control card R/D)
- US Patent 5,646,496 (used in control card R/D and 1 Vp-p feedback interface)
- US Patent 6,118,241 (used in control card simple dynamic braking)
- US Patent 8,154,228 (Dynamic Braking For Electric Motors)
- US Patent 8,214,063 (Auto-tune of a Control System Based on Frequency Response)

Patents referring to fieldbus functions are listed in the matching fieldbus manual.

Technical changes which improve the performance of the device may be made without prior notice.

Printed in the United States of America

This document is the intellectual property of Kollmorgen. All rights reserved. No part of this work may be reproduced in any form (by photocopying, microfilm or any other method) or stored, processed, copied or distributed by electronic means without the written permission of Kollmorgen.

1 Table of Contents

| | | |
|----------|---|-----------|
| 1 | Table of Contents | 3 |
| 2 | Welcome to AKD PDMM User Guide | 25 |
| 3 | AKD PDMM Models | 26 |
| 4 | Software Setup | 27 |
| 4.1 | Install and Start the KAS IDE | 27 |
| 4.2 | KAS IDE Project View | 29 |
| 4.3 | Configuring the Drives and Remote I/O | 29 |
| 4.4 | Where To Go From Here | 31 |
| 5 | Initial Drive Setup | 33 |
| 5.1 | Initial Drive Setup | 34 |
| 5.2 | AKD PDMM Display Codes | 35 |
| 5.3 | AKD PDMM Setup Wizard | 36 |
| 5.4 | Settings | 37 |
| 5.4.1 | Navigation Tree | 38 |
| 5.4.2 | Settings View | 38 |
| 6 | Configuring Drive Power | 40 |
| 6.1 | Power | 41 |
| 6.1.1 | Drive Setup for Power and Bus | 41 |
| 6.2 | Operating Voltage | 41 |
| 6.3 | Direct DC Mains Operation | 41 |
| 6.4 | Regeneration | 43 |
| 6.4.1 | Overview | 43 |
| 6.4.2 | Regen Resistor Options | 43 |
| 6.4.3 | Calculating Motor Peak Energy and Regen Resistor Size | 44 |
| 6.4.4 | Selecting a Compatible Regen Resistor | 46 |
| 6.4.5 | Configuring Regen Parameter Values | 47 |
| 7 | Configuring Motor Settings | 49 |
| 7.1 | Motor | 50 |
| 7.1.1 | Overview | 50 |
| 7.1.2 | Motor Setup | 50 |
| 7.1.3 | Using the Motor View | 50 |
| 7.1.4 | Selecting a Motor | 51 |
| 7.1.5 | Configuring Custom Motors | 51 |
| 7.1.5.1 | Validating Motor Parameters | 52 |
| 7.1.6 | Motor Derating | 52 |
| 7.1.7 | Motor Temperature | 54 |
| 7.2 | Feedback 1 | 54 |
| 7.2.1 | Overview | 55 |
| 7.2.2 | Using Feedback Options | 55 |
| 7.2.2.1 | Auto | 55 |
| 7.2.2.2 | None | 55 |
| 7.2.2.3 | Incremental Encoder | 56 |
| 7.2.2.4 | Sine Encoder | 56 |
| 7.2.2.5 | Endat 2.1, Endat 2.2 | 56 |

| | | |
|------------|--|-----------|
| 7.2.2.6 | BiSS Analog | 56 |
| 7.2.2.7 | Hiperface, Hiperface DSL | 56 |
| 7.2.2.8 | Resolver | 56 |
| 7.2.2.9 | SFD, SFD3 | 56 |
| 7.2.2.10 | Tamagawa | 56 |
| 7.2.2.11 | Halls Only | 56 |
| 7.2.3 | Wake and Shake Overview | 56 |
| 7.3 | Extended Multiturn | 57 |
| 7.4 | Non-Plug and Play Feedback Devices | 58 |
| 7.4.1 | Parameters | 58 |
| 7.4.2 | Calculations | 58 |
| 7.4.2.0.1 | Current Loop | 58 |
| 7.4.2.0.2 | Velocity Loop | 58 |
| 7.4.2.0.3 | Slider Tuning | 58 |
| 7.4.2.0.4 | Input - Motor Data | 58 |
| 7.4.2.0.5 | Constants | 58 |
| 7.4.2.0.6 | Output - Control Loop Gains | 59 |
| 7.5 | Foldback | 59 |
| 7.5.1 | Drive Foldback | 59 |
| 7.5.2 | Setting up motor foldback | 59 |
| 7.5.3 | Setting Fault and Warning Levels | 59 |
| 7.5.4 | Motor Peak Current Time | 60 |
| 7.5.5 | Motor Foldback Ramp | 60 |
| 7.5.6 | Motor Recovery | 61 |
| 7.5.7 | Overall Foldback | 61 |
| 7.6 | Motor I2t algorithm | 62 |
| 7.6.1 | Parameters | 62 |
| 7.6.2 | Functional description of the motor I2t algorithm | 62 |
| 7.6.3 | Details about IL.MI2T | 64 |
| 7.7 | Brake | 64 |
| 7.8 | Using Position Capture | 64 |
| 7.8.1 | Overview | 64 |
| 7.8.2 | Configuring Position Capture | 65 |
| 7.8.2.0.1 | Setting the Capture Source (CAP0.FBSOURCE, CAP1.FBSOURCE (pg 1)) | 65 |
| 7.8.2.0.2 | Setting the Trigger Source (CAP0.TRIGGER, CAP1.TRIGGER (pg 1)) | 65 |
| 7.8.2.0.3 | Setting the Capture Mode (CAP0.MODE, CAP1.MODE) | 65 |
| 7.8.2.0.4 | Arming and Retrieving the Capture Value (CAP0.T, CAP1.T) | 66 |
| 7.8.2.0.5 | Setting the Capture Edge (CAP0.EDGE, CAP1.EDGE) | 66 |
| 7.8.2.0.6 | Setting the Pre-Condition Event: (CAP0.EVENT, CAP1.EVENT) | 66 |
| 7.8.2.0.7 | Setting up a Pre-Condition for complex capture | 66 |
| 7.8.3 | Kollmorgen Test Reports | 66 |
| 8 | Using AKD PDMM in a Vertical Axis | 68 |
| 9 | Configuring with Linear Motors | 70 |
| 9.1 | Connecting a DDL Motor to an AKD PDMM Drive | 70 |

| | | |
|-------------|--|-----------|
| 10 | Configuring General Drive Settings | 73 |
| 10.1 | Limits | 74 |
| 10.1.1 | Limits | 74 |
| 10.2 | Compare Engines | 75 |
| 10.2.1 | Configuring Position Compare | 75 |
| 10.2.1.1 | Order of Configuration | 75 |
| 10.2.1.2 | Setting Compare Sources (CMPx.SOURCE (→ p. 1)) | 75 |
| 10.2.2 | Setting Compare Setpoints (CMPx.SETPOINT (→ p. 1)) | 76 |
| 10.3 | Enable/Disable | 77 |
| 10.3.1 | Enable Modes | 77 |
| 10.3.1.0.1 | Hardware Enable Mode | 77 |
| 10.3.1.0.2 | Software Enable Default | 77 |
| 10.3.2 | Disable Modes | 77 |
| 10.3.3 | Drive Status | 78 |
| 10.3.4 | Controlled Stop | 79 |
| 10.3.5 | More/Less Button | 79 |
| 10.4 | Controlled Stop | 80 |
| 10.5 | Dynamic Braking | 82 |
| 10.5.1 | Drive Regeneration | 82 |
| 10.5.1.0.1 | AKD-M00306 to AKD-M00606 | 82 |
| 10.5.1.0.2 | AKD-M01206 to AKD-Mzzz07 | 82 |
| 10.6 | Emergency Stop | 82 |
| 10.6.1 | Stop / Emergency Stop / Emergency Off | 82 |
| 10.6.1.1 | Stop | 83 |
| 10.6.1.2 | Emergency Stop | 84 |
| 10.6.1.3 | Emergency Off | 84 |
| 10.7 | Under Voltage Fault Behavior | 84 |
| 10.7.1 | Current Loop Gain Scheduling | 84 |
| 10.7.1.1 | Using the Gain Scheduling View in EWW | 85 |
| 10.7.1.1.1 | Using the Terminal View for Gains Scheduling | 86 |
| 10.7.1.2 | Rules for Populating Gain Scheduling Table | 87 |
| 11 | Creating Motion | 88 |
| 11.1 | Service Motion | 89 |
| 11.2 | Motion Profile Table | 90 |
| 11.2.1 | Grid | 91 |
| 11.2.2 | Graphical Representation | 91 |
| 11.2.3 | Control Buttons | 92 |
| 11.2.3.1 | Import Table Data | 93 |
| 11.2.3.2 | Importing data from Preset Table option | 93 |
| 11.2.3.3 | Importing data from an external CSV file | 94 |
| 11.2.4 | Creating Motion Profile Tables | 94 |
| 11.2.4.1 | Example of a motion profile table | 94 |
| 11.2.4.2 | Motion Profile Table Restrictions | 95 |
| 11.2.4.3 | Different methods of motion table motion tasking | 96 |
| 11.2.4.3.1 | General motion profile table explanations | 96 |

| | | |
|-------------|---|------------|
| 11.2.4.4 | Use of IL.KACCFF with Motion Tables | 97 |
| 11.2.4.5 | Profile Table Interpolation Motion Task | 97 |
| 11.2.4.6 | OneToOne interpolation motion task | 97 |
| 11.2.4.7 | Setting up a motion profile motion task | 98 |
| 11.2.4.8 | Drive reaction on impossible motion tasks | 98 |
| 11.2.4.8.1 | OneToOne interpolation limitations | 98 |
| 11.2.4.9 | Profile Table interpolation limitations | 99 |
| 11.2.4.9.1 | Starting from velocity 0 without change-on-the-fly to a following motion task | 99 |
| 11.2.4.9.2 | During a change on the fly condition | 99 |
| 11.2.4.9.3 | Movement to the same direction | 99 |
| 11.2.4.9.4 | Movement in different directions | 100 |
| 12 | Tuning Your System | 102 |
| 12.1 | Introduction | 103 |
| 12.2 | Slider Tuning | 103 |
| 12.2.1 | Load Inertia | 103 |
| 12.2.2 | Gentle, Medium, and Stiff | 104 |
| 12.2.3 | The Slider | 104 |
| 12.3 | Using the Performance Servo Tuner | 104 |
| 12.3.0.1 | Overview | 104 |
| 12.3.0.2 | Using the PST | 105 |
| 12.3.0.2.1 | Saving and Emailing Bode Plots | 106 |
| 12.3.0.2.2 | Importing a Frequency Response | 107 |
| 12.3.0.3 | Measurement Options | 107 |
| 12.3.0.3.1 | Using Manual Excitation Levels | 107 |
| 12.3.0.4 | Taking a Bode Measurement without the PST | 108 |
| 12.3.1 | Using the Performance Servo Tuner: Advanced | 108 |
| 12.3.1.1 | Typical Cases for Advanced PST Use | 108 |
| 12.3.1.2 | PST Options | 113 |
| 12.3.1.3 | Measurement Options | 114 |
| 12.3.1.4 | Plot Options | 119 |
| 12.3.1.5 | Resizing Bode Plots | 122 |
| 12.4 | Tuning Guide | 132 |
| 12.4.1 | Overview | 132 |
| 12.4.2 | Determining Tuning Criteria | 133 |
| 12.4.3 | Before You Tune | 133 |
| 12.4.4 | Closed Loop Tuning Methods | 133 |
| 12.4.4.1 | Tuning the Velocity Loop | 133 |
| 12.4.4.2 | Tuning the Position Loop | 136 |
| 12.4.5 | Torque Feedforward Tuning Methods | 137 |
| 12.4.5.1 | Shape Based Feedforward Tuning | 137 |
| 12.4.6 | Using Anti-Resonance Filters | 137 |
| 12.4.6.1 | Types of Anti-Resonance Filters | 138 |
| 12.4.6.2 | Biquad Calculations | 145 |
| 12.4.6.3 | Common Uses Of Anti-Resonance Filters | 149 |
| 12.5 | Cogging Compensation | 149 |

| | |
|--|------------|
| 12.5.1 Controlling Cogging Compensation | 149 |
| 12.5.1.1 Enabling | 149 |
| 12.5.1.2 Cogging Compensation View | 150 |
| 12.5.1.3 Teaching | 151 |
| 12.5.1.4 Validating Cogging Compensation | 152 |
| 13 Scope | 154 |
| 13.1 Overview | 154 |
| 13.2 Using the Scope | 154 |
| 13.2.1 Scope Channels Tab | 154 |
| 13.2.1.1 Source Column | 154 |
| 13.2.1.2 Color Column | 155 |
| 13.2.1.3 Hide Column | 155 |
| 13.2.1.4 Y-Axis Column | 155 |
| 13.2.1.5 Filter and Filter Frequency Column | 155 |
| 13.2.2 Measure Tab | 155 |
| 13.2.2.1 AC RMS and True RMS | 155 |
| 13.2.3 Scope Time-base and Trigger Tab | 155 |
| 13.2.3.1 Scope Time-base and Trigger, More View | 156 |
| 13.2.3.2 Trigger Type | 157 |
| 13.2.3.3 Trigger Position | 158 |
| 13.2.3.4 Trigger Value | 158 |
| 13.2.3.5 Effects of Recorder Gap | 159 |
| 13.2.3.6 Trigger Slope | 160 |
| 13.3 User Traces | 161 |
| 13.3.1 Add and Configure a User Trace | 161 |
| 13.3.2 Saving User Traces | 163 |
| 13.4 Scope Settings | 163 |
| 13.4.1 Load a setting (preset) to Scope screen | 164 |
| 13.4.2 Create a new preset | 164 |
| 13.4.3 Save or delete preset | 164 |
| 13.4.4 Import preset | 164 |
| 13.4.5 Export preset | 165 |
| 13.4.6 Scope axis scaling and zooming | 166 |
| 13.4.7 Manual range per axis | 167 |
| 13.4.8 Unit display on Y axis | 167 |
| 14 Using Parameters and the Terminal Screen | 168 |
| 14.1 Terminal | 169 |
| 14.1.1 Overview | 169 |
| 14.1.2 Using the Terminal | 169 |
| 14.1.3 Macros | 170 |
| 14.1.3.0.1 Creating a Macro from Terminal commands | 170 |
| 14.1.3.0.2 Macro Editor | 172 |
| 14.2 Parameter List | 173 |
| 14.3 Parameter Load/Save | 175 |
| 14.3.1 Loading an Incompatible File | 175 |
| 15 Faults and Warnings | 176 |
| 15.1 Fault and Warning Messages | 177 |

| | |
|---|------------|
| 15.2 AKD PDMM Errors and Alarms | 193 |
| 15.3 Additional error and alarm messages AKD PDMM | 194 |
| 15.3.1 Additional error and alarm messages AKD PDMM-M | 194 |
| 15.3.2 Errors | 194 |
| 15.3.3 Alarms | 198 |
| 15.4 Clearing Faults | 199 |
| 15.5 Parameter and Command Error Messages | 199 |
| 15.6 Unknown Fault | 203 |
| 15.6.1 Remedies | 203 |
| 16 Troubleshooting the AKD PDMM | 204 |
| 17 Connection Diagrams | 206 |
| 17.1 Connection diagram AKD-M00306, AKD-M00606 | 207 |
| 17.2 Connection diagram AKD-M01206 | 208 |
| 17.3 Connection diagram AKD-M02406, AKD PDMM-M00307 to AKD PDMM-M02407 | 209 |
| 17.4 24 V auxiliary supply (X1) | 210 |
| 17.4.1 AKD-M003 to 024, connector X1 | 210 |
| 17.4.2 AKD-M048, connector X1 | 210 |
| 17.5 Motor Power Connection (X2) | 211 |
| 17.5.1 AKD-M003 to 024, power connector X2 | 213 |
| 17.5.2 AKD-M048, power connector X2 | 213 |
| 17.6 External regen resistor (X3) | 214 |
| 17.6.1 AKD-M003 to 024, regen connector X3 | 214 |
| 17.6.2 AKD-M048, regen connector X3 | 216 |
| 17.7 DC Bus link (X3, X14) | 216 |
| 17.7.1 DC Bus topology with Y connectors (24 A max.) | 218 |
| 17.7.2 DC Bus topology with busbar | 218 |
| 17.8 Mains supply connection (X3, X4) | 219 |
| 17.8.1 Three phase connection (all AKD PDMM types) | 221 |
| 17.8.2 Single/Dual phase connection (AKD-M0010600306 to AKD-M01206 only) | 221 |
| 17.9 I/O Connection | 222 |
| 17.9.1 Overview | 222 |
| 17.9.1.1 I/O connectors X7 and X8 (all AKD PDMM variants) | 222 |
| 17.9.1.2 I/O connectors X35 and X36 (AKD PDMM only) | 224 |
| 17.10 Analog Output (X8) | 225 |
| 17.11 Analog Input (X8) | 226 |
| 17.12 Command encoder signal connection | 227 |
| 17.12.1 Incremental encoder input 5 V (X9) | 227 |
| 17.12.2 Incremental encoder input 24 V (X7) | 227 |
| 17.12.3 Encoder with EnDat 2.2 input 5 V (X9) | 229 |
| 17.13 Pulse / Direction signal connection | 229 |
| 17.13.1 Pulse / Direction input 5 V (X9) | 230 |
| 17.13.2 Pulse / Direction Input 5 V to 24 V (X7) | 230 |
| 17.14 CW / CCW signal connection | 230 |
| 17.14.1 CW / CCW input 5 V (X9) | 231 |
| 17.14.2 CW / CCW input 24 V (X7) | 231 |
| 17.15 Feedback connector (X10) | 231 |
| 17.16 Feedback connector (X9) | 232 |

| | | |
|-------------|---|------------|
| 17.17 | Feedback connector (X7) | 233 |
| 18 | AKD PDMM Firmware | 234 |
| 18.1 | Check AKD PDMM Drive Firmware | 234 |
| 18.2 | Download AKD PDMM Drive Firmware | 234 |
| 18.3 | AKD PDMM Firmware Update | 234 |
| 19 | Parameter and Command Reference Guide | 236 |
| 19.1 | About the Parameter and Command Reference Guide | 249 |
| 19.1.1 | Parameter and Command Naming Conventions | 251 |
| 19.1.2 | Summary of Parameters and Commands | 252 |
| 19.2 | AOUT Parameters | 269 |
| 19.2.1 | AOUT.CUTOFF | 270 |
| 19.2.2 | AOUT.DEBUGADDR | 271 |
| 19.2.3 | AOUT.DEBUGDATATYPE | 272 |
| 19.2.4 | AOUT.DEBUGSCALE | 273 |
| 19.2.5 | AOUT.ISCALE | 274 |
| 19.2.6 | AOUT.MODE | 275 |
| 19.2.7 | AOUT.OFFSET | 276 |
| 19.2.8 | AOUT.PSCALE | 277 |
| 19.2.9 | AOUT.VALUE | 278 |
| 19.2.10 | AOUT.VALUEU | 279 |
| 19.2.11 | AOUT.VSCALE | 280 |
| 19.3 | BODE Parameters | 281 |
| 19.3.1 | BODE.EXCITEGAP | 282 |
| 19.3.2 | BODE.FREQ | 283 |
| 19.3.3 | BODE.IAMP | 284 |
| 19.3.4 | BODE.IFLIMIT | 285 |
| 19.3.5 | BODE.IFTHRESH | 286 |
| 19.3.6 | BODE.INJECTPOINT | 287 |
| 19.3.7 | BODE.MODE | 288 |
| 19.3.8 | BODE.MODETIMER | 291 |
| 19.3.9 | BODE.PRBDEPTH | 293 |
| 19.3.10 | BODE.VAMP | 294 |
| 19.3.11 | BODE.VFLIMIT | 295 |
| 19.3.12 | BODE.VFTHRESH | 296 |
| 19.4 | CMP Parameters | 297 |
| 19.4.1 | CMPx.ADVANCE | 298 |
| 19.4.2 | CMPx.ARM | 299 |
| 19.4.3 | CMPx.MODBOUND1 | 300 |
| 19.4.4 | CMPx.MODBOUND2 | 302 |
| 19.4.5 | CMPx.MODE | 304 |
| 19.4.6 | CMPx.MODEN | 306 |
| 19.4.7 | CMPx.MODVALUE | 307 |
| 19.4.8 | CMPx.OUTMASK | 308 |
| 19.4.9 | CMPx.SETPOINT | 309 |
| 19.4.10 | CMPx.SOURCE | 311 |
| 19.4.11 | CMPx.SOURCEVALUE | 312 |
| 19.4.12 | CMPx.STATE | 313 |

| | |
|---|------------|
| 19.4.13 CMPx.WIDTH | 314 |
| 19.4.14 CMPx.WIDTHTYPE | 316 |
| 19.5 COGCOMP Parameters | 318 |
| 19.5.1 COGCOMP.CORRECTIONTABLE | 319 |
| 19.5.2 COGCOMP.CORRECTIONVALUE | 320 |
| 19.5.3 COGCOMP.EN | 321 |
| 19.5.4 COGCOMP.LOAD | 322 |
| 19.5.5 COGCOMP.RANGEHIGH | 323 |
| 19.5.6 COGCOMP.RANGELOW | 324 |
| 19.5.7 COGCOMP.SAVE | 325 |
| 19.5.8 COGCOMP.TEACH | 326 |
| 19.5.9 COGCOMP.V | 327 |
| 19.6 CS Parameters | 328 |
| 19.6.1 CS.DEC | 329 |
| 19.6.2 CS.STATE | 330 |
| 19.6.3 CS.TO | 331 |
| 19.6.4 CS.VTHRESH | 332 |
| 19.7 DIN Parameters | 333 |
| 19.7.1 DIN.HCMD1 TO DIN.HCMD4 | 334 |
| 19.7.2 DIN.LCMD1 to DIN.LCMD4 | 335 |
| 19.7.3 DIN.ROTARY | 336 |
| 19.7.4 DIN.STATES | 337 |
| 19.7.5 DIN1.FILTER TO DIN7.FILTER | 338 |
| 19.7.6 DIN1.INV to DIN7.INV | 339 |
| 19.7.7 DIN1.MODE TO DIN7.MODE | 340 |
| 19.7.8 DIN1.PARAM TO DIN7.PARAM | 342 |
| 19.7.9 DIN1.STATE TO DIN7.STATE | 344 |
| 19.7.10 DIN9.STATE to DIN11.STATE | 345 |
| 19.8 DIO Parameters | 346 |
| 19.8.1 DIO9.INV to DIO11.INV | 347 |
| 19.8.2 DIO9.DIR to DIO11.DIR | 348 |
| 19.9 DOUT Parameters | 349 |
| 19.9.1 DOUT.CTRL | 350 |
| 19.9.2 DOUT.RELAYMODE | 351 |
| 19.9.3 DOUT.STATES | 352 |
| 19.9.4 DOUT1.MODE and DOUT2.MODE | 353 |
| 19.9.5 DOUT1.PARAM AND DOUT2.PARAM | 355 |
| 19.9.6 DOUT1.STATE AND DOUT2.STATE | 357 |
| 19.9.7 DOUT1.STATEU AND DOUT2.STATEU | 358 |
| 19.9.8 DOUT9.MODE to DOUT11.MODE | 359 |
| 19.9.9 DOUT9.PARAM to DOUT11.MODE | 360 |
| 19.9.10 DOUT9.STATE to DOUT11.STATE | 361 |
| 19.9.11 DOUT9.STATEU to DOUT11.STATEU | 362 |
| 19.10 DRV Parameters | 363 |
| 19.10.1 DRV.ACC | 366 |
| 19.10.2 DRV.ACTIVE | 368 |
| 19.10.3 DRV.BLINKDISPLAY | 369 |

| | |
|--|-----|
| 19.10.4 DRV.BOOTTIME | 370 |
| 19.10.5 DRV.CLRASHDUMP | 371 |
| 19.10.6 DRV.CLRFAULTHIST | 372 |
| 19.10.7 DRV.CLRFAULTS | 373 |
| 19.10.8 DRV.CMDDELAY | 374 |
| 19.10.9 DRV.CMDSOURCE | 375 |
| 19.10.10 DRV.CRASHDUMP | 376 |
| 19.10.11 DRV.CUSTOMIDENTIFIER | 377 |
| 19.10.12 DRV.DBILIMIT | 378 |
| 19.10.13 DRV.DEC | 379 |
| 19.10.14 DRV.DIFVAR | 381 |
| 19.10.15 DRV.DIR | 382 |
| 19.10.16 DRV.DIS | 384 |
| 19.10.17 DRV.DISMODE | 385 |
| 19.10.18 DRV.DISSOURCES | 386 |
| 19.10.19 DRV.DISSOURCESMASK | 387 |
| 19.10.20 DRV.DISTO | 388 |
| 19.10.21 DRV.DOWNLOADALLOWED | 389 |
| 19.10.22 DRV.EMUECHECKSPEED | 390 |
| 19.10.23 DRV.EMUEDIR | 391 |
| 19.10.24 DRV.EMUEMODE | 392 |
| 19.10.25 DRV.EMUEMTURN | 394 |
| 19.10.26 DRV.EMUEPULSEWIDTH | 395 |
| 19.10.27 DRV.EMUERES | 396 |
| 19.10.28 DRV.EMUESTEPCMD | 397 |
| 19.10.29 DRV.EMUESTEPCMDPIN | 398 |
| 19.10.30 DRV.EMUESTEPCMDPOUT | 399 |
| 19.10.31 DRV.EMUEZOFFSET | 400 |
| 19.10.32 DRV.EN | 401 |
| 19.10.33 DRV.ENDEFAULT | 402 |
| 19.10.34 DRV.ERRORLIST | 403 |
| 19.10.35 DRV.FAULTDISPLAYMODE | 404 |
| 19.10.36 DRV.FAULTHIST | 405 |
| 19.10.37 DRV.FAULT1 to DRV.FAULT10 | 406 |
| 19.10.38 DRV.FAULTS | 407 |
| 19.10.39 DRV.HANDWHEELSRC | 408 |
| 19.10.40 DRV.HELP | 409 |
| 19.10.41 DRV.HELPALL | 410 |
| 19.10.42 DRV.HWENABLE | 411 |
| 19.10.43 DRV.HWENDELAY | 412 |
| 19.10.44 DRV.HWENMODE | 413 |
| 19.10.45 DRV.ICONT | 414 |
| 19.10.46 DRV.INFO | 415 |
| 19.10.47 DRV.IPEAK | 417 |
| 19.10.48 DRV.IZERO | 418 |
| 19.10.49 DRV.LIST | 419 |
| 19.10.50 DRV.LOGICVOLTS | 420 |

| | |
|---|------------|
| 19.10.51 DRV.MEMADDR | 421 |
| 19.10.52 DRV.MEMDATA | 422 |
| 19.10.53 DRV.MOTIONDISSOURCES | 423 |
| 19.10.54 DRV.NAME | 424 |
| 19.10.55 DRV.NVCHECK | 425 |
| 19.10.56 DRV.NVLIST | 426 |
| 19.10.57 DRV.NVLOAD | 427 |
| 19.10.58 DRV.NVSAVE | 428 |
| 19.10.59 DRV.ONTIME | 429 |
| 19.10.60 DRV.OPMODE | 430 |
| 19.10.61 DRV.READFORMAT | 431 |
| 19.10.62 DRV.REBOOT | 432 |
| 19.10.63 DRV.RSTVAR | 433 |
| 19.10.64 DRV.RUNTIME | 434 |
| 19.10.65 DRV.SETUPREQBITS | 435 |
| 19.10.66 DRV.SETUPREQLIST | 437 |
| 19.10.67 DRV.STOP | 438 |
| 19.10.68 DRV.TEMPERATURES | 439 |
| 19.10.69 DRV.TYPE | 440 |
| 19.10.70 DRV.VER | 441 |
| 19.10.71 DRV.VERIMAGE | 442 |
| 19.10.72 DRV.WARNING1 to DRV.WARNING3 | 443 |
| 19.10.73 DRV.WARNINGS | 444 |
| 19.10.74 DRV.ZERO | 445 |
| 19.11 DS402 Parameters | 446 |
| 19.11.1 DS402.1ADDPOSFCFEED | 447 |
| 19.11.2 DS402.1ADDPOSFCSHAFTREV | 448 |
| 19.11.3 DS402.3ADDPOSFCFEED | 449 |
| 19.11.4 DS402.3ADDPOSFCSHAFTREV | 450 |
| 19.12 FAULT Parameters | 451 |
| 19.12.1 FAULTx.ACTION | 452 |
| 19.12.2 FAULT570.ACTION | 453 |
| 19.13 FB1 Parameters | 454 |
| 19.13.1 FB1.BISSBITS | 456 |
| 19.13.2 FB1.CALTHRESH | 457 |
| 19.13.3 FB1.DIAG | 458 |
| 19.13.4 FB1.ENCRES | 459 |
| 19.13.5 FB1.EXTENDEDMULTITURN | 460 |
| 19.13.6 FB1.FAULTS | 461 |
| 19.13.7 FB1.HALLSTATE | 463 |
| 19.13.8 FB1.HALLSTATEU | 464 |
| 19.13.9 FB1.HALLSTATEV | 465 |
| 19.13.10 FB1.HALLSTATEW | 466 |
| 19.13.11 FB1.IDENTIFIED | 467 |
| 19.13.12 FB1.INITPSAVED | 468 |
| 19.13.13 FB1.INITPSTATUS | 469 |
| 19.13.14 FB1.INITPWINDOW | 470 |

| | |
|---|------------|
| 19.13.15 FB1.INITSIGNED | 471 |
| 19.13.16 FB1.LASTIDENTIFIED | 472 |
| 19.13.17 FB1.MECHPOS | 473 |
| 19.13.18 FB1.MEMVER | 474 |
| 19.13.19 FB1.MOTORPHASE | 475 |
| 19.13.20 FB1.MOTORPOLES | 476 |
| 19.13.21 FB1.OFFSET | 477 |
| 19.13.22 FB1.ORIGIN | 478 |
| 19.13.23 FB1.P | 480 |
| 19.13.24 FB1.PDIR | 481 |
| 19.13.25 FB1.PFIND | 482 |
| 19.13.26 FB1.PFINDCMDU | 483 |
| 19.13.27 FB1.PIN | 484 |
| 19.13.28 FB1.POFFSET | 485 |
| 19.13.29 FB1.POLES | 486 |
| 19.13.30 FB1.POUT | 487 |
| 19.13.31 FB1.PSCALE | 488 |
| 19.13.32 FB1.PUNIT | 489 |
| 19.13.33 FB1.RESKTR | 490 |
| 19.13.34 FB1.RESREFPHASE | 491 |
| 19.13.35 FB1.SELECT | 492 |
| 19.13.36 FB1.TRACKINGCAL | 494 |
| 19.13.37 FB1.USERBYTE0 to FB1.USERBYTE7 | 495 |
| 19.13.38 FB1.USERDWORD0 to FB1.USERDWORD1 | 497 |
| 19.13.39 FB1.USERWORD0 to FB1.USERWORD3 | 499 |
| 19.14 FB3 Parameters | 501 |
| 19.14.1 FB3.DIR | 502 |
| 19.14.2 FB3.ENCRES | 503 |
| 19.14.3 FB3.MODE | 504 |
| 19.14.4 FB3.MOTORPHASE | 505 |
| 19.14.5 FB3.MOTORPOLES | 506 |
| 19.14.6 FB3.P | 507 |
| 19.14.7 FB3.PDIR | 508 |
| 19.14.8 FB3.PIN | 509 |
| 19.14.9 FB3.POFFSET | 510 |
| 19.14.10 FB3.POUT | 511 |
| 19.14.11 FB3.PUNIT | 512 |
| 19.15 FBUS Parameters | 513 |
| 19.15.1 FBUS.PARAM1 TO FBUS.PARAM10 | 514 |
| 19.15.2 FBUS.PLLSTATE | 517 |
| 19.15.3 FBUS.PLLTHRESH | 518 |
| 19.15.4 FBUS.PROTECTION | 519 |
| 19.15.5 FBUS.REMOTE | 521 |
| 19.15.6 FBUS.SAMPLEPERIOD | 522 |
| 19.15.7 FBUS.STATE | 523 |
| 19.15.8 FBUS.SYNACQUIREWND | 524 |
| 19.15.9 FBUS.SYNCACT | 525 |

| | |
|------------------------------------|------------|
| 19.15.10 FBUS.SYNCDIST | 526 |
| 19.15.11 FBUS.SYNCKLOCKWND | 527 |
| 19.15.12 FBUS.SYNCWND | 528 |
| 19.15.13 FBUS.TYPE | 529 |
| 19.16 GUI Parameters | 530 |
| 19.16.1 GUI.DISPLAY | 531 |
| 19.16.2 GUI.DISPLAYS | 532 |
| 19.16.3 GUI.PARAM01 | 533 |
| 19.16.4 GUI.PARAM02 | 534 |
| 19.16.5 GUI.PARAM03 | 535 |
| 19.16.6 GUI.PARAM04 | 536 |
| 19.16.7 GUI.PARAM05 | 537 |
| 19.16.8 GUI.PARAM06 | 538 |
| 19.16.9 GUI.PARAM07 | 539 |
| 19.16.10 GUI.PARAM08 | 540 |
| 19.16.11 GUI.PARAM09 | 541 |
| 19.16.12 GUI.PARAM10 | 542 |
| 19.17 HWLS Parameters | 543 |
| 19.17.1 HWLS.NEGSTATE | 544 |
| 19.17.2 HWLS.POSSTATE | 545 |
| 19.18 IL Parameters | 546 |
| 19.18.1 IL.BUSFF | 548 |
| 19.18.2 IL.CMD | 549 |
| 19.18.3 IL.CMDACC | 550 |
| 19.18.4 IL.CMDU | 551 |
| 19.18.5 IL.DCMD | 552 |
| 19.18.6 IL.DFB | 553 |
| 19.18.7 IL.DIFOLD | 554 |
| 19.18.8 IL.FB | 555 |
| 19.18.9 IL.FBSOURCE | 556 |
| 19.18.10 IL.FF | 557 |
| 19.18.11 IL.FOLDFTHRESH | 558 |
| 19.18.12 IL.FOLDFTHRESHU | 559 |
| 19.18.13 IL.FOLDWTHRESH | 560 |
| 19.18.14 IL.FRCTION | 561 |
| 19.18.15 IL.IFOLD | 562 |
| 19.18.16 IL.IUFB | 563 |
| 19.18.17 IL.IVFB | 564 |
| 19.18.18 IL.KACFF | 565 |
| 19.18.19 IL.KBUSFF | 566 |
| 19.18.20 IL.KP | 567 |
| 19.18.21 IL.KPDRATIO | 568 |
| 19.18.22 IL.KPLOOKUPINDEX | 569 |
| 19.18.23 IL.KPLOOKUPVALUE | 570 |
| 19.18.24 IL.KPLOOKUPVALUES | 571 |
| 19.18.25 IL.KPSOURCE | 572 |
| 19.18.26 IL.KVFF | 573 |

| | |
|--------------------------------------|------------|
| 19.18.27 IL.LIMITN | 574 |
| 19.18.28 IL.LIMITP | 575 |
| 19.18.29 IL.MFOLDD | 576 |
| 19.18.30 IL.MFOLDR | 577 |
| 19.18.31 IL.MFOLDT | 578 |
| 19.18.32 IL.MI2T | 579 |
| 19.18.33 IL.MI2TWTRESH | 580 |
| 19.18.34 IL.MIFOLD | 581 |
| 19.18.35 IL.MIMODE | 582 |
| 19.18.36 IL.OFFSET | 583 |
| 19.18.37 IL.VCMD | 584 |
| 19.18.38 IL.VLIMIT | 585 |
| 19.18.39 IL.VUFB | 586 |
| 19.18.40 IL.VVFB | 587 |
| 19.19 LOAD Parameters | 588 |
| 19.19.1 LOAD.INERTIA | 589 |
| 19.20 MOTOR Parameters | 590 |
| 19.20.1 MOTOR.AUTOSET | 592 |
| 19.20.2 MOTOR.BRAKE | 594 |
| 19.20.3 MOTOR.BRAKEIMM | 595 |
| 19.20.4 MOTOR.BRAKERLS | 596 |
| 19.20.5 MOTOR.BRAKESTATE | 597 |
| 19.20.6 MOTOR.CTF0 | 598 |
| 19.20.7 MOTOR.FIELDWEAKENING | 599 |
| 19.20.8 MOTOR.ICONT | 600 |
| 19.20.9 MOTOR.IDDATAVALID | 601 |
| 19.20.10 MOTOR.IDMAX | 602 |
| 19.20.11 MOTOR.IMID | 603 |
| 19.20.12 MOTOR.IMTR | 604 |
| 19.20.13 MOTOR.INERTIA | 606 |
| 19.20.14 MOTOR.IPEAK | 607 |
| 19.20.15 MOTOR.KE | 608 |
| 19.20.16 MOTOR.KT | 609 |
| 19.20.17 MOTOR.LDLL | 610 |
| 19.20.18 MOTOR.LISAT | 611 |
| 19.20.19 MOTOR.LQLL | 612 |
| 19.20.20 MOTOR.NAME | 613 |
| 19.20.21 MOTOR.PHASE | 614 |
| 19.20.22 MOTOR.PHSADV1 | 615 |
| 19.20.23 MOTOR.PHSADV2 | 616 |
| 19.20.24 MOTOR.PITCH | 617 |
| 19.20.25 MOTOR.POLES | 618 |
| 19.20.26 MOTOR.R | 619 |
| 19.20.27 MOTOR.RTYPE | 620 |
| 19.20.28 MOTOR.SUPPORTEDPARAMS | 621 |
| 19.20.29 MOTOR.TBRAKEAPP | 622 |
| 19.20.30 MOTOR.TBRAKERLS | 623 |

| | |
|-----------------------------------|------------|
| 19.20.31 MOTOR.TBRAKETO | 624 |
| 19.20.32 MOTOR.TEMP | 625 |
| 19.20.33 MOTOR.TEMPC | 626 |
| 19.20.34 MOTOR.TEMPFALT | 627 |
| 19.20.35 MOTOR.TEMPWARN | 628 |
| 19.20.36 MOTOR.TYPE | 629 |
| 19.20.37 MOTOR.VMAX | 631 |
| 19.20.38 MOTOR.VOLTMAX | 632 |
| 19.20.39 MOTOR.VOLTMIN | 633 |
| 19.20.40 MOTOR.VOLTRATED | 634 |
| 19.20.41 MOTOR.VRATED | 635 |
| 19.21 PL Parameters | 636 |
| 19.21.1 PL.CMD | 637 |
| 19.21.2 PL.ERR | 638 |
| 19.21.3 PL.ERRFTHRESH | 639 |
| 19.21.4 PL.ERRMODE | 641 |
| 19.21.5 PL.ERRWTHRESH | 643 |
| 19.21.6 PL.FB | 644 |
| 19.21.7 PL.FBSOURCE | 645 |
| 19.21.8 PL.FILTERTIME | 646 |
| 19.21.9 PL.INTINMAX | 647 |
| 19.21.10 PL.INTOUTMAX | 649 |
| 19.21.11 PL.KI | 651 |
| 19.21.12 PL.KITHRESH | 652 |
| 19.21.13 PL.KP | 653 |
| 19.21.14 PL.PDELAY | 654 |
| 19.22 REC Parameters | 655 |
| 19.22.1 REC.ACTIVE | 656 |
| 19.22.2 REC.CH1 to REC.CH6 | 657 |
| 19.22.3 REC.DONE | 658 |
| 19.22.4 REC.GAP | 659 |
| 19.22.5 REC.NUMPOINTS | 660 |
| 19.22.6 REC.OFF | 661 |
| 19.22.7 REC.RECPRMLIST | 662 |
| 19.22.8 REC.RETRIEVE | 663 |
| 19.22.9 REC.RETRIEVEDATA | 664 |
| 19.22.10 REC.RETRIEVEFRMT | 666 |
| 19.22.11 REC.RETRIEVEHDR | 667 |
| 19.22.12 REC.RETRIEVESIZE | 668 |
| 19.22.13 REC.STOPTYPE | 669 |
| 19.22.14 REC.TRIG | 670 |
| 19.22.15 REC.TRIGPARAM | 671 |
| 19.22.16 REC.TRIGPOS | 672 |
| 19.22.17 REC.TRIGPRMLIST | 674 |
| 19.22.18 REC.TRIGSLOPE | 675 |
| 19.22.19 REC.TRIGTYPE | 676 |
| 19.22.20 REC.TRIGVAL | 677 |

| | |
|--|------------|
| 19.23 REGEN Parameters | 678 |
| 19.23.1 REGEN.POWER | 679 |
| 19.23.2 REGEN.POWERFILTERED | 680 |
| 19.23.3 REGEN.REXT | 681 |
| 19.23.4 REGEN.TEXT | 682 |
| 19.23.5 REGEN.TYPE | 683 |
| 19.23.6 REGEN.WATTEXT | 684 |
| 19.24 SM Parameters | 685 |
| 19.24.1 SM.ACC | 686 |
| 19.24.2 SM.DEC | 687 |
| 19.24.3 SM.I1 | 688 |
| 19.24.4 SM.I2 | 689 |
| 19.24.5 SM.MODE | 690 |
| 19.24.6 SM.MOVE | 693 |
| 19.24.7 SM.T1 | 694 |
| 19.24.8 SM.T2 | 695 |
| 19.24.9 SM.V1 | 696 |
| 19.24.10 SM.V2 | 697 |
| 19.25 STO Parameters | 698 |
| 19.25.1 STO.STATE | 699 |
| 19.26 SWLS Parameters | 700 |
| 19.26.1 SWLS.EN | 701 |
| 19.26.2 SWLS.LIMIT0 | 702 |
| 19.26.3 SWLS.LIMIT1 | 703 |
| 19.26.4 SWLS.STATE | 704 |
| 19.27 TEMP Parameters | 705 |
| 19.27.1 TEMP.CONTROL | 706 |
| 19.27.2 TEMP.POWER1 to TEMP.POWER3 | 707 |
| 19.28 USER Parameters | 708 |
| 19.28.1 USER.INT1 to USER.INT24 | 709 |
| 19.29 VBUS Parameters | 710 |
| 19.29.1 VBUS.HALFVOLT | 711 |
| 19.29.2 VBUS.OVFTHRESH | 712 |
| 19.29.3 VBUS.OVWTHRESH | 713 |
| 19.29.4 VBUS.RMSLIMIT | 714 |
| 19.29.5 VBUS.UVFTHRESH | 715 |
| 19.29.6 VBUS.UVMODE | 716 |
| 19.29.7 VBUS.UVWTHRESH | 717 |
| 19.29.8 VBUS.VALUE | 718 |
| 19.30 VL Parameters | 719 |
| 19.30.1 VL.ARPF1 TO VL.ARPF4 | 720 |
| 19.30.2 VL.ARPQ1 TO VL.ARPQ4 | 722 |
| 19.30.3 VL.ARTYPE1 TO VL.ARTYPE4 | 724 |
| 19.30.4 VL.ARZF1 TO VL.ARZF4 | 725 |
| 19.30.5 VL.ARZQ1 TO VL.ARZQ4 | 726 |
| 19.30.6 VL.BUSFF | 728 |
| 19.30.7 VL.CMD | 729 |

| | |
|---|------------|
| 19.30.8 VL.CMDU | 730 |
| 19.30.9 VL.ERR | 731 |
| 19.30.10 VL.FB | 732 |
| 19.30.11 VL.FBFILTER | 733 |
| 19.30.12 VL.FBSOURCE | 734 |
| 19.30.13 VL.FBUNFILTERED | 735 |
| 19.30.14 VL.FF | 736 |
| 19.30.15 VL.FFDELAY | 737 |
| 19.30.16 VL.GENMODE | 738 |
| 19.30.17 VL.KBUSFF | 739 |
| 19.30.18 VL.KI | 740 |
| 19.30.19 VL.KIMODE | 741 |
| 19.30.20 VL.KO | 742 |
| 19.30.21 VL.KP | 743 |
| 19.30.22 VL.KVFF | 745 |
| 19.30.23 VL.LIMITN | 746 |
| 19.30.24 VL.LIMITP | 748 |
| 19.30.25 VL.LMJR | 749 |
| 19.30.26 VL.MODEL | 750 |
| 19.30.27 VL.OBSBW | 751 |
| 19.30.28 VL.OBSMODE | 752 |
| 19.30.29 VL.THRESH | 753 |
| 19.30.30 VL.VFTHRESH | 755 |
| 19.31 WS Parameters | 756 |
| 19.31.1 WS.ARM | 757 |
| 19.31.2 WS.CHECKMODE | 758 |
| 19.31.3 WS.CHECKT | 759 |
| 19.31.4 WS.CHECKV | 760 |
| 19.31.5 WS.DISARM | 761 |
| 19.31.6 WS.DISTMAX | 762 |
| 19.31.7 WS.DISTMIN | 763 |
| 19.31.8 WS.FREQ | 764 |
| 19.31.9 WS.IMAX | 765 |
| 19.31.10 WS.MODE | 766 |
| 19.31.11 WS.NUMLOOPS | 767 |
| 19.31.12 WS.STATE | 768 |
| 19.31.13 WS.T | 769 |
| 19.31.14 WS.TDELAY1 | 770 |
| 19.31.15 WS.TDELAY2 | 771 |
| 19.31.16 WS.TDELAY3 | 772 |
| 19.31.17 WS.TDELAY4 | 773 |
| 19.31.18 WS.TIRAMP | 774 |
| 19.31.19 WS.TSTANDSTILL | 775 |
| 19.31.20 WS.VTHRESH | 776 |
| 19.32 EtherCAT Profile | 778 |
| 19.32.1 Slave Register | 779 |
| 19.32.2 AL Event (Interrupt Event) and Interrupt Enable | 780 |

| | | |
|--------------|--|------------|
| 19.32.2.1 | Interrupt Enable Register (Address 0x0204:0x0205) | 780 |
| 19.32.2.2 | AL Event Request (Address 0x0220:0x0221) | 781 |
| 19.32.3 | Phase Run-Up | 782 |
| 19.32.3.1 | AL Control (Address 0x0120:0x0121) | 782 |
| 19.32.3.2 | AL Status (Address 0x0130:0x0131) | 782 |
| 19.32.3.3 | AL Status Code (Address 0x0134:0x0135) | 783 |
| 19.32.3.4 | EtherCAT communication phases | 783 |
| 19.32.4 | CANopen over EtherCAT (CoE) State Machine | 785 |
| 19.32.4.1 | Status Description | 785 |
| 19.32.4.2 | Commands in the Control Word | 786 |
| 19.32.4.3 | State Machine Bits (status word) | 788 |
| 19.32.5 | Fixed PDO Mappings | 789 |
| 19.32.6 | Flexible PDO Mappings | 791 |
| 19.32.6.1 | Example: Flexible PDO Mapping | 793 |
| 19.32.7 | Supported Cyclical Setpoint and Actual Values | 797 |
| 19.32.8 | Supported Operation Modes | 799 |
| 19.32.9 | Adjusting EtherCAT Cycle Time | 799 |
| 19.32.10 | Maximum Cycle Times depending on operation mode | 799 |
| 19.32.11 | Synchronization | 801 |
| 19.32.11.1 | Synchronization behavior with distributed clocks (DC) enabled | 801 |
| 19.32.11.2 | Synchronization behavior with distributed clocks (DC) disabled | 801 |
| 19.32.12 | Latch Control Word and Latch Status Word | 802 |
| 19.32.13 | Mailbox Handling | 804 |
| 19.32.13.1 | Mailbox Output | 805 |
| 19.32.13.2 | Mailbox Input | 807 |
| 19.32.13.3 | Example: Mailbox Access | 809 |
| 19.32.14 | EEProm Content | 810 |
| 19.33 | Appendix | 812 |
| 19.33.1 | CANopen Emergency Messages and Error Codes | 812 |
| 19.33.2 | Object Dictionary | 818 |
| 19.33.2.1 | Float Scaling | 818 |
| 19.33.2.2 | Effectiveness of PDO set-points | 818 |
| 19.33.2.3 | Communication SDOs | 818 |
| 19.33.2.4 | Manufacturer specific SDOs | 822 |
| 19.33.2.5 | Profile specific SDOs | 839 |
| 19.33.3 | Object descriptions | 844 |
| 19.33.3.1 | Object 1000h: Device Type (DS301) | 844 |
| 19.33.3.2 | Object 1001h: Error register (DS301) | 845 |
| 19.33.3.3 | Object 1002h: Manufacturer Status Register (DS301) | 846 |
| 19.33.3.4 | Object 1003h: Predefined Error Field (DS301) | 847 |
| 19.33.3.5 | Object 1005h: COB-ID of the SYNC Message (DS301) | 848 |
| 19.33.3.6 | Object 1006h: Communication Cycle Period (DS301) | 848 |
| 19.33.3.7 | Object 1008h: Manufacturer Device Name (DS301) | 849 |
| 19.33.3.8 | Object 1009h: Manufacturer Hardware Version | 849 |
| 19.33.3.9 | Object 100Ah: Manufacturer Software Version (DS301) | 849 |
| 19.33.3.10 | Object 100Ch: Guard Time (DS301)Response monitoring | 850 |
| 19.33.3.11 | Object 100Dh: Lifetime Factor (DS301) | 850 |

| | | |
|------------|---|-----|
| 19.33.3.12 | Object 1010h: Store Parameters (DS301) | 851 |
| 19.33.3.13 | Object 1011h: Restore Default Parameters DS301 | 852 |
| 19.33.3.14 | Object 1012h: COB-ID of the Time Stamp (DS301) | 853 |
| 19.33.3.15 | Object 1014h: COB-ID for Emergency Message (DS301) | 853 |
| 19.33.3.16 | Object 1016h: Consumer Heartbeat Time | 854 |
| 19.33.3.17 | Object 1017h: Producer Heartbeat Time | 855 |
| 19.33.3.18 | Object 1018h: Identity Object (DS301) | 855 |
| 19.33.3.19 | Object 1026h: OS Prompt | 857 |
| 19.33.3.20 | Objects 1400-1403h: 1st - 4th RxPDO communication parameter (DS301) | 858 |
| 19.33.3.21 | Objects 1600-1603h: 1st - 4th RxPDO mapping parameter (DS301) | 859 |
| 19.33.3.22 | Objects 1800-1803h: 1st - 4th TxPDO communication parameter (DS301) | 861 |
| 19.33.3.23 | Objects 1A00-1A03h: 1st - 4th TxPDO mapping parameter (DS301) | 864 |
| 19.33.3.24 | Object 1C12h: RxPDO assign (DS301) | 865 |
| 19.33.3.25 | Object 1C13h: TxPDO assign (DS301) | 866 |
| 19.33.3.26 | Object 2000h: System Warnings | 867 |
| 19.33.3.27 | Object 2001h: System Faults | 867 |
| 19.33.3.28 | Object 2002h: Manufacturer status bytes | 868 |
| 19.33.3.29 | Object 2011h: DRV.RUNTIME in seconds | 869 |
| 19.33.3.30 | Object 2012h: Fault history: Fault numbers | 870 |
| 19.33.3.31 | Object 2013h: Fault history: Time stamps | 871 |
| 19.33.3.32 | Object 2014-2017h: 1st-4th Mask 1 to 4 for Transmit-PDO | 872 |
| 19.33.3.33 | Object 2018h: Firmware Version | 873 |
| 19.33.3.34 | Object 2026h: ASCII Channel | 875 |
| 19.33.3.35 | Object 204Ch: PV Scaling Factor | 876 |
| 19.33.3.36 | Object 2071h: Target Current | 877 |
| 19.33.3.37 | Object 2077h: Current Actual Value | 877 |
| 19.33.3.38 | Object 20A0h: Latch position 1, positive edge | 877 |
| 19.33.3.39 | Object 20A1h: Latch position 1, negative edge | 879 |
| 19.33.3.40 | Object 20A2h: Latch position 2, positive edge | 879 |
| 19.33.3.41 | Object 20A3h: Latch position 2, negative edge | 880 |
| 19.33.3.42 | Object 20A4h: Latch Control Register | 880 |
| 19.33.3.43 | Object 20A5h: Latch Status Register | 881 |
| 19.33.3.44 | Object 20A6h: Latch position 1, positive or negative edge | 881 |
| 19.33.3.45 | Object 20A7h: Latch position 2, positive or negative edge | 883 |
| 19.33.3.46 | Object 20B8h: Reset of changed input information | 883 |
| 19.33.3.47 | Object 345Ah: Brake Control | 884 |
| 19.33.3.48 | Object 3474h: Parameters for digital inputs | 887 |
| 19.33.3.49 | Object 3475h: Parameters for digital outputs | 888 |
| 19.33.3.50 | Object 3496h: Fieldbus synchronization parameters | 889 |
| 19.33.3.51 | Object 6040h: Control word (DS402) | 891 |
| 19.33.3.52 | Object 6041h: Status word (DS402) | 892 |
| 19.33.3.53 | Object 605Ah: Quick stop option code (DS402) | 895 |
| 19.33.3.54 | Object 6060h: Modes of Operation (DS402) | 896 |
| 19.33.3.55 | Object 6061h: Modes of Operation Display (DS402) | 897 |
| 19.33.3.56 | Object 6063h: position actual value* (DS402) | 897 |
| 19.33.3.57 | Object 6064h: position actual value (DS402) | 897 |
| 19.33.3.58 | Object 6065h: Following error window | 898 |

| | | |
|-------------|--|------------|
| 19.33.3.59 | Object 606Ch: Velocity actual value (DS402) | 898 |
| 19.33.3.60 | Object 6071h: Target torque (DS402) | 899 |
| 19.33.3.61 | Object 6073h: Max current (DS402) | 899 |
| 19.33.3.62 | Object 6077h: Torque actual value (DS402) | 899 |
| 19.33.3.63 | Object 607Ah: Target position (DS402) | 900 |
| 19.33.3.64 | Object 607Ch: Homing offset (DS402) | 900 |
| 19.33.3.65 | Object 607Dh: Software position limit (DS402) | 901 |
| 19.33.3.66 | Object 6081h: Profile velocity (DS402) | 902 |
| 19.33.3.67 | Object 6083h: Profile acceleration (DS402) | 902 |
| 19.33.3.68 | Object 6084h: Profile deceleration (DS402) | 902 |
| 19.33.3.69 | Object 6087h Torque slope (DS402) | 904 |
| 19.33.3.70 | Object 608Fh: Position encoder resolution (DS402) | 904 |
| 19.33.3.71 | Object 6091h: Gear Ratio (DS402) | 905 |
| 19.33.3.72 | Object 6092h: Feed constant (DS402) | 906 |
| 19.33.3.73 | Object 6098h: Homing method (DS402) | 907 |
| 19.33.3.74 | Object 6099h: Homing speeds (DS402) | 908 |
| 19.33.3.75 | Object 609Ah: Homing acceleration (DS402) | 910 |
| 19.33.3.76 | Object 60B1h: Velocity Offset | 910 |
| 19.33.3.77 | Object 60B2h: Torque Offset | 910 |
| 19.33.3.78 | Object 60B8h: Touch probe function | 911 |
| 19.33.3.79 | Object 60B9h: Touch probe status | 913 |
| 19.33.3.80 | Object 60BAh: Touch probe 1 positive edge | 914 |
| 19.33.3.81 | Object 60BBh: Touch probe 1 negative edge | 914 |
| 19.33.3.82 | Object 60BCh: Touch probe 2 positive edge | 914 |
| 19.33.3.83 | Object 60BDh: Touch probe 2 negative edge | 915 |
| 19.33.3.84 | Object 60C0h: Interpolation sub mode select | 915 |
| 19.33.3.85 | Object 60C1h: Interpolation data record | 916 |
| 19.33.3.86 | Object 60C2h: Interpolation time period | 917 |
| 19.33.3.87 | Object 60C4h: Interpolation data configuration | 918 |
| 19.33.3.88 | Object 60D0h: Touch probe source | 920 |
| 19.33.3.89 | Object 60E0h: Positive Torque Limit Value | 921 |
| 19.33.3.90 | Object 60E1h: Negative Torque Limit Value | 921 |
| 19.33.3.91 | Object 60E4h: Additional position actual value | 922 |
| 19.33.3.92 | Object 60E8h: Additional gear ratio – motor shaft revolutions | 923 |
| 19.33.3.93 | Object 60E9h: Additional feed constant – feed | 924 |
| 19.33.3.94 | Object 60EDh: Additional gear ratio – driving shaft revolutions | 925 |
| 19.33.3.95 | Object 60EEh: Additional feed constant - driving shaft revolutions | 926 |
| 19.33.3.96 | Object 60F4h: Following error actual value (DS402) | 927 |
| 19.33.3.97 | Object 60FCh: Position demand internal value (DS402) | 928 |
| 19.33.3.98 | Object 60FDh: Digital inputs (DS402) | 928 |
| 19.33.3.99 | Object 60FEh: Digital outputs (DS402) | 929 |
| 19.33.3.100 | Object 60FFh: Target velocity (DS402) | 930 |
| 19.33.3.101 | Object 6502h: Supported drive modes (DS402) | 930 |
| 20 | EtherCAT Profile | 932 |
| 20.1 | Slave Register | 933 |
| 20.2 | AL Event (Interrupt Event) and Interrupt Enable | 934 |
| 20.2.1 | Interrupt Enable Register (Address 0x0204:0x0205) | 934 |

| | |
|--|-------------|
| 20.2.2 AL Event Request (Address 0x0220:0x0221) | 935 |
| 20.3 Phase Run-Up | 936 |
| 20.3.1 AL Control (Address 0x0120:0x0121) | 936 |
| 20.3.2 AL Status (Address 0x0130:0x0131) | 936 |
| 20.3.3 AL Status Code (Address 0x0134:0x0135) | 937 |
| 20.3.4 EtherCAT communication phases | 937 |
| 20.4 CANopen over EtherCAT (CoE) State Machine | 939 |
| 20.4.1 Status Description | 939 |
| 20.4.2 Commands in the Control Word | 940 |
| 20.4.3 State Machine Bits (status word) | 942 |
| 20.5 Fixed PDO Mappings | 943 |
| 20.6 Flexible PDO Mappings | 945 |
| 20.6.1 Example: Flexible PDO Mapping | 947 |
| 20.7 Supported Cyclical Setpoint and Actual Values | 951 |
| 20.8 Supported Operation Modes | 953 |
| 20.9 Adjusting EtherCAT Cycle Time | 953 |
| 20.10 Maximum Cycle Times depending on operation mode | 953 |
| 20.11 Synchronization | 955 |
| 20.11.1 Synchronization behavior with distributed clocks (DC) enabled | 955 |
| 20.11.2 Synchronization behavior with distributed clocks (DC) disabled | 955 |
| 20.12 Latch Control Word and Latch Status Word | 956 |
| 20.13 Mailbox Handling | 958 |
| 20.13.1 Mailbox Output | 959 |
| 20.13.2 Mailbox Input | 961 |
| 20.13.3 Example: Mailbox Access | 963 |
| 20.14 Important Configuration Parameters | 964 |
| 20.15 EEPROM Content | 965 |
| 21 Appendix | 968 |
| 21.1 CANopen Emergency Messages and Error Codes | 968 |
| 21.2 Object Dictionary | 974 |
| 21.2.1 Float Scaling | 974 |
| 21.2.2 Effectiveness of PDO set-points | 974 |
| 21.2.3 Communication SDOs | 974 |
| 21.2.4 Manufacturer specific SDOs | 978 |
| 21.2.5 Profile specific SDOs | 995 |
| 21.3 Object descriptions | 1000 |
| 21.3.1 Object 1000h: Device Type (DS301) | 1000 |
| 21.3.2 Object 1001h: Error register (DS301) | 1001 |
| 21.3.3 Object 1002h: Manufacturer Status Register (DS301) | 1002 |
| 21.3.4 Object 1003h: Predefined Error Field (DS301) | 1003 |
| 21.3.5 Object 1005h: COB-ID of the SYNC Message (DS301) | 1004 |
| 21.3.6 Object 1006h: Communication Cycle Period (DS301) | 1004 |
| 21.3.7 Object 1008h: Manufacturer Device Name (DS301) | 1005 |
| 21.3.8 Object 1009h: Manufacturer Hardware Version | 1005 |
| 21.3.9 Object 100Ah: Manufacturer Software Version (DS301) | 1005 |
| 21.3.10 Object 100Ch: Guard Time (DS301)Response monitoring | 1006 |
| 21.3.11 Object 100Dh: Lifetime Factor (DS301) | 1006 |

| | |
|---|------|
| 21.3.12 Object 1010h: Store Parameters (DS301) | 1007 |
| 21.3.13 Object 1011h: Restore Default Parameters DS301 | 1008 |
| 21.3.14 Object 1012h: COB-ID of the Time Stamp (DS301) | 1009 |
| 21.3.15 Object 1014h: COB-ID for Emergency Message (DS301) | 1009 |
| 21.3.16 Object 1016h: Consumer Heartbeat Time | 1010 |
| 21.3.17 Object 1017h: Producer Heartbeat Time | 1011 |
| 21.3.18 Object 1018h: Identity Object (DS301) | 1011 |
| 21.3.19 Object 1026h: OS Prompt | 1013 |
| 21.3.20 Objects 1400-1403h: 1st - 4th RxPDO communication parameter (DS301) | 1014 |
| 21.3.21 Objects 1600-1603h: 1st - 4th RxPDO mapping parameter (DS301) | 1015 |
| 21.3.22 Objects 1800-1803h: 1st - 4th TxPDO communication parameter (DS301) | 1017 |
| 21.3.23 Objects 1A00-1A03h: 1st - 4th TxPDO mapping parameter (DS301) | 1020 |
| 21.3.24 Object 1C12h: RxPDO assign (DS301) | 1021 |
| 21.3.25 Object 1C13h: TxPDO assign (DS301) | 1022 |
| 21.3.26 Object 2000h: System Warnings | 1023 |
| 21.3.27 Object 2001h: System Faults | 1023 |
| 21.3.28 Object 2002h: Manufacturer status bytes | 1024 |
| 21.3.29 Object 2011h: DRV.RUNTIME in seconds | 1025 |
| 21.3.30 Object 2012h: Fault history: Fault numbers | 1026 |
| 21.3.31 Object 2013h: Fault history: Time stamps | 1027 |
| 21.3.32 Object 2014-2017h: 1st-4th Mask 1 to 4 for Transmit-PDO | 1028 |
| 21.3.33 Object 2018h: Firmware Version | 1029 |
| 21.3.34 Object 2026h: ASCII Channel | 1031 |
| 21.3.35 Object 204Ch: PV Scaling Factor | 1032 |
| 21.3.36 Object 2071h: Target Current | 1033 |
| 21.3.37 Object 2077h: Current Actual Value | 1033 |
| 21.3.38 Object 20A0h: Latch position 1, positive edge | 1033 |
| 21.3.39 Object 20A1h: Latch position 1, negative edge | 1035 |
| 21.3.40 Object 20A2h: Latch position 2, positive edge | 1035 |
| 21.3.41 Object 20A3h: Latch position 2, negative edge | 1036 |
| 21.3.42 Object 20A4h: Latch Control Register | 1036 |
| 21.3.43 Object 20A5h: Latch Status Register | 1038 |
| 21.3.44 Object 20A6h: Latch position 1, positive or negative edge | 1038 |
| 21.3.45 Object 20A7h: Latch position 2, positive or negative edge | 1040 |
| 21.3.46 Object 20B8h: Reset of changed input information | 1040 |
| 21.3.47 Object 345Ah: Brake Control | 1041 |
| 21.3.48 Object 3474h: Parameters for digital inputs | 1044 |
| 21.3.49 Object 3475h: Parameters for digital outputs | 1045 |
| 21.3.50 Object 3496h: Fieldbus synchronization parameters | 1046 |
| 21.3.51 Object 6040h: Control word (DS402) | 1048 |
| 21.3.52 Object 6041h: Status word (DS402) | 1049 |
| 21.3.53 Object 605Ah: Quick stop option code (DS402) | 1052 |
| 21.3.54 Object 6060h: Modes of Operation (DS402) | 1053 |
| 21.3.55 Object 6061h: Modes of Operation Display (DS402) | 1054 |
| 21.3.56 Object 6063h: position actual value* (DS402) | 1054 |
| 21.3.57 Object 6064h: position actual value (DS402) | 1054 |
| 21.3.58 Object 6065h: Following error window | 1055 |

| | |
|--|-------------|
| 21.3.59 Object 606Ch: Velocity actual value (DS402) | 1055 |
| 21.3.60 Object 6071h: Target torque (DS402) | 1056 |
| 21.3.61 Object 6073h: Max current (DS402) | 1056 |
| 21.3.62 Object 6077h: Torque actual value (DS402) | 1056 |
| 21.3.63 Object 607Ah: Target position (DS402) | 1057 |
| 21.3.64 Object 607Ch: Homing offset (DS402) | 1057 |
| 21.3.65 Object 607Dh: Software position limit (DS402) | 1058 |
| 21.3.66 Object 6081h: Profile velocity (DS402) | 1059 |
| 21.3.67 Object 6083h: Profile acceleration (DS402) | 1059 |
| 21.3.68 Object 6084h: Profile deceleration (DS402) | 1059 |
| 21.3.69 Object 6087h Torque slope (DS402) | 1061 |
| 21.3.70 Object 608Fh: Position encoder resolution (DS402) | 1061 |
| 21.3.71 Object 6091h: Gear Ratio (DS402) | 1062 |
| 21.3.72 Object 6092h: Feed constant (DS402) | 1064 |
| 21.3.73 Object 6098h: Homing method (DS402) | 1065 |
| 21.3.74 Object 6099h: Homing speeds (DS402) | 1066 |
| 21.3.75 Object 609Ah: Homing acceleration (DS402) | 1068 |
| 21.3.76 Object 60B1h: Velocity Offset | 1068 |
| 21.3.77 Object 60B2h: Torque Offset | 1068 |
| 21.3.78 Object 60B8h: Touch probe function | 1069 |
| 21.3.79 Object 60B9h: Touch probe status | 1071 |
| 21.3.80 Object 60BAh: Touch probe 1 positive edge | 1072 |
| 21.3.81 Object 60BBh: Touch probe 1 negative edge | 1072 |
| 21.3.82 Object 60BCh: Touch probe 2 positive edge | 1072 |
| 21.3.83 Object 60BDh: Touch probe 2 negative edge | 1073 |
| 21.3.84 Object 60C0h: Interpolation sub mode select | 1073 |
| 21.3.85 Object 60C1h: Interpolation data record | 1074 |
| 21.3.86 Object 60C2h: Interpolation time period | 1075 |
| 21.3.87 Object 60C4h: Interpolation data configuration | 1076 |
| 21.3.88 Object 60D0h: Touch probe source | 1078 |
| 21.3.89 Object 60E0h: Positive Torque Limit Value | 1079 |
| 21.3.90 Object 60E1h: Negative Torque Limit Value | 1079 |
| 21.3.91 Object 60E4h: Additional position actual value | 1080 |
| 21.3.92 Object 60E8h: Additional gear ratio – motor shaft revolutions | 1081 |
| 21.3.93 Object 60E9h: Additional feed constant – feed | 1082 |
| 21.3.94 Object 60EDh: Additional gear ratio – driving shaft revolutions | 1083 |
| 21.3.95 Object 60EEh: Additional feed constant - driving shaft revolutions | 1084 |
| 21.3.96 Object 60F4h: Following error actual value (DS402) | 1085 |
| 21.3.97 Object 60FCh: Position demand internal value (DS402) | 1086 |
| 21.3.98 Object 60FDh: Digital inputs (DS402) | 1086 |
| 21.3.99 Object 60FEh: Digital outputs (DS402) | 1087 |
| 21.3.100 Object 60FFh: Target velocity (DS402) | 1088 |
| 21.3.101 Object 6502h: Supported drive modes (DS402) | 1088 |
| 22 Glossary | 1090 |
| 23 Index | 1102 |

2 Welcome to AKD PDMM User Guide

Welcome to the AKD PDMM User Guide, a guide for installing, setting up, and using an AKD PDMM. Complete product documentation in pdf or online help format is included on your installation disk. Product manuals are also posted on the [Kollmorgen website](#) and are updated on the website as needed between releases. Check the website for the most current manuals.

The documentation available on the Kollmorgen website includes the following:

Related documents for the AKD PDMM series include:

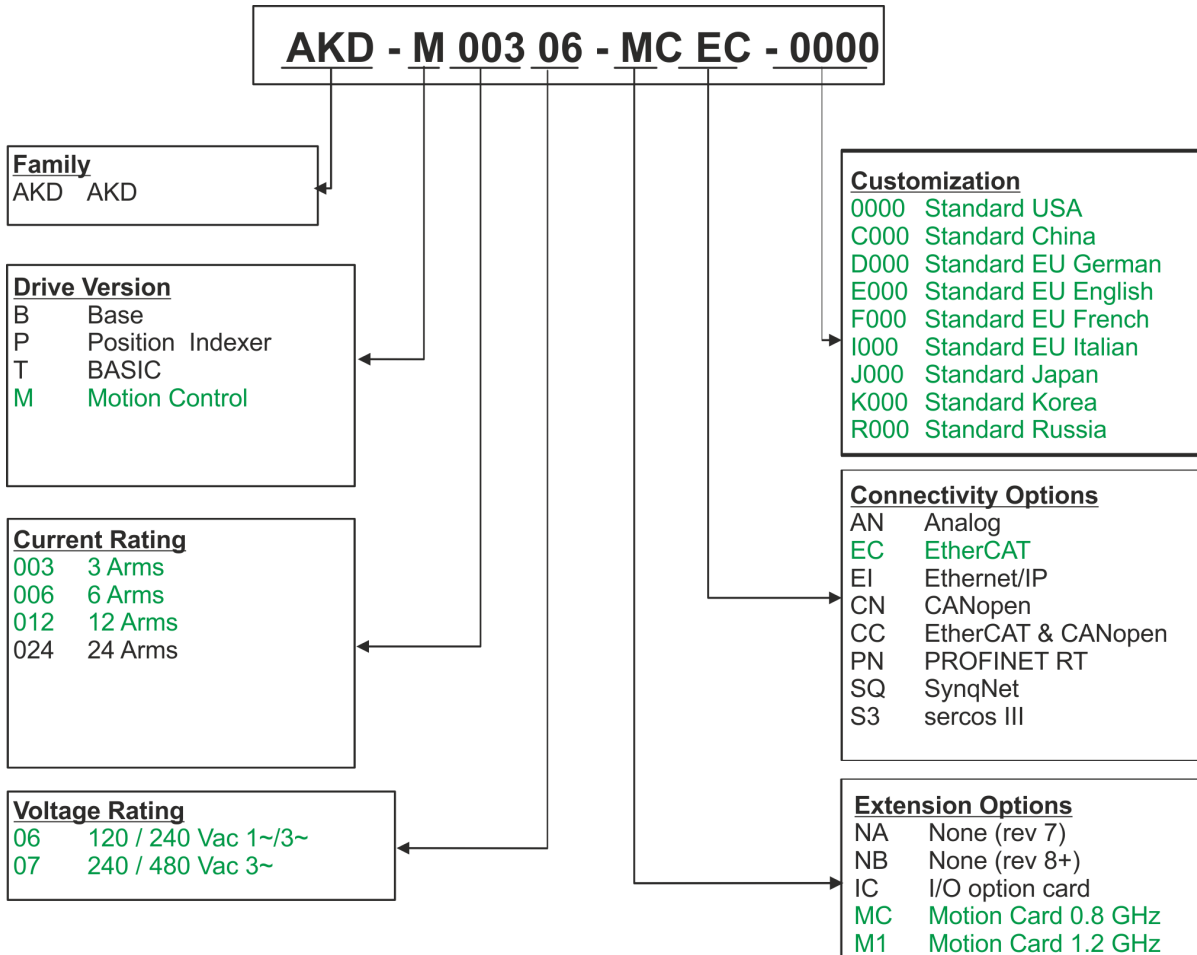
- *AKD PDMM Quick Start* (also provided in hard copy). This guide provides instructions for initial drive setup and connection to a network.
- *AKD PDMM Installation Manual* (also provided in hard copy for EU customers). This manual provides instructions for installation and drive setup.
- *AKD PDMM Users Manual*. This manual describes how to use your drive in common applications. It also provides tips for maximizing your system performance with the AKD PDMM.
- *Accessories Manual*. This manual provides documentation for accessories like cables and regen resistors used with AKD PDMM. Regional versions of this manual exist.

3 AKD PDMM Models

AKD PDMM drive models are available in a variety of combinations of features. The part number identifies the features included in your model.

NOTE
 AKD PDMM is only available for EtherCAT field busses.

The figure below shows part number identification for drive features. Features available in the AKD PDMM are marked bold green.



4 Software Setup

4.1 Install and Start the KAS IDE

Once the Kollmorgen Automation Suite Integrated Development Environment (KAS IDE) installation is complete, click the IDE icon to launch the program.

NOTE

The KVB software used for developing an HMI display does not install with the KAS IDE and must be installed separately.

Begin a new project in the KAS IDE by selecting **File > New**. This will launch the **Add a New Controller** window.

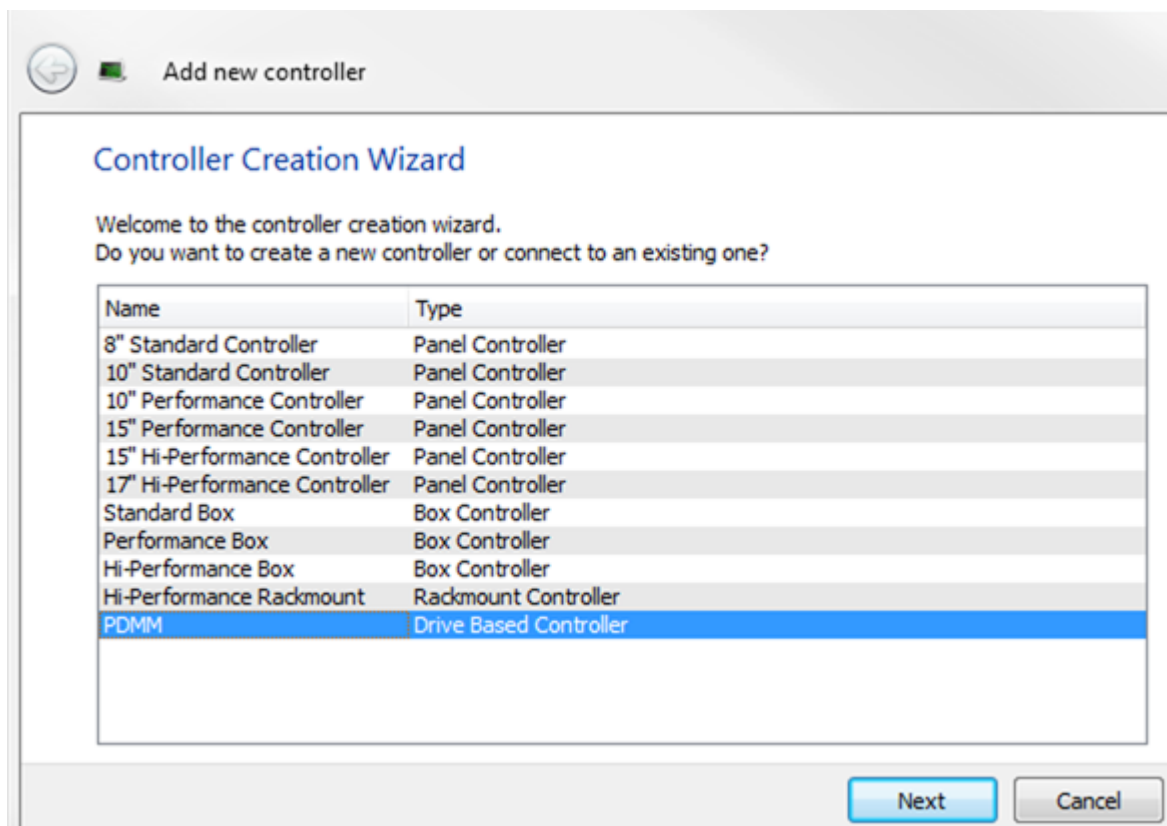


Figure 4-1: Add a New Controller

Select your AKD PDMM model from the list and click Next. You will then be prompted to select an application template:

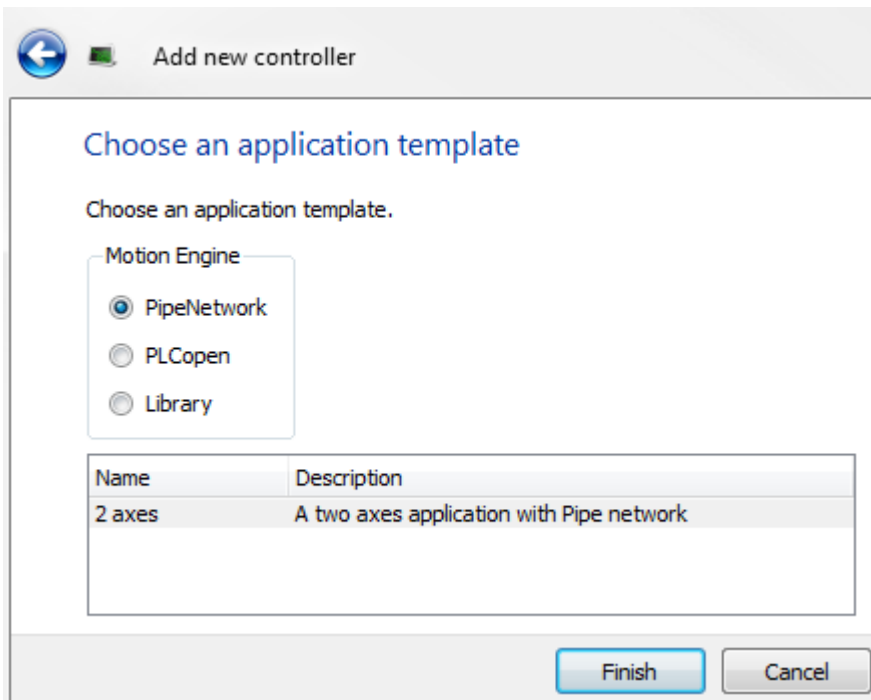


Figure 4-2: Select Application Template

Choose from PipeNetwork, PLCopen, or Library, and click Finish. The controller will then be added to the project view.

To associate the project with the IP address of the PDMM controller, right click on the Controller option in the Project View.

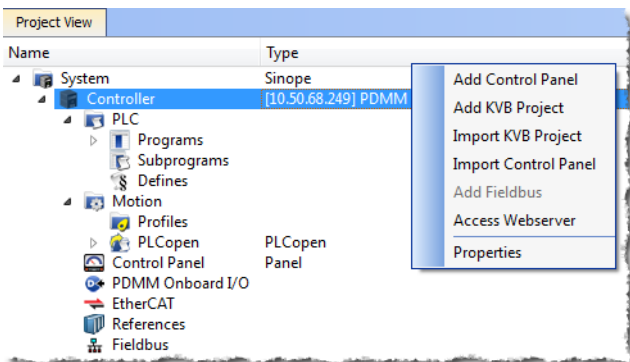


Figure 4-3: Controller Selection

Select Properties and the following screen appears:

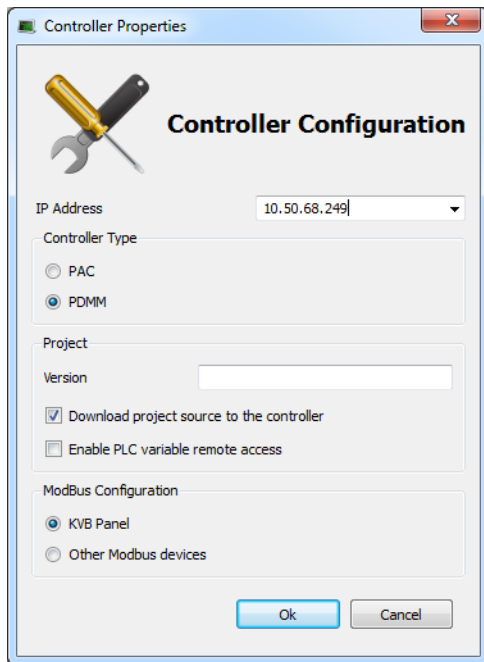


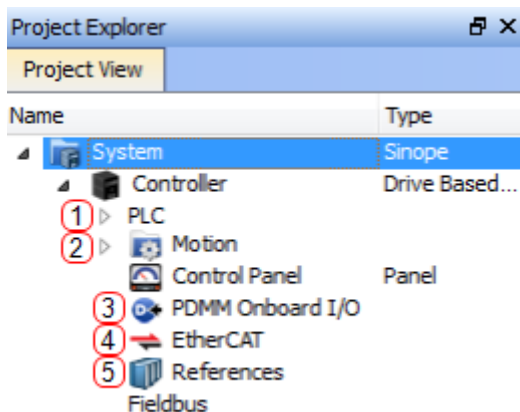
Figure 4-4: Controller Properties

Type in the IP address of the PDMM, set the Controller Type to PDMM and click OK.

4.2 KAS IDE Project View

The KAS IDE contains tools for configuring the EtherCAT network, setting up and tuning the drives, adding and configuring HMIs, and creating a PLC program. For more complete information see the KAS IDE User Guide.

Once a project (new or saved) is opened, the user can open a variety of items to build a project from the Project Explorer:



1. Create PLC Programs
2. Add Motion
3. Configure Resident I/O
4. Configure Drives and Remote I/O
5. Add Additional Reference Documents

Figure 4-5: Project View

4.3 Configuring the Drives and Remote I/O

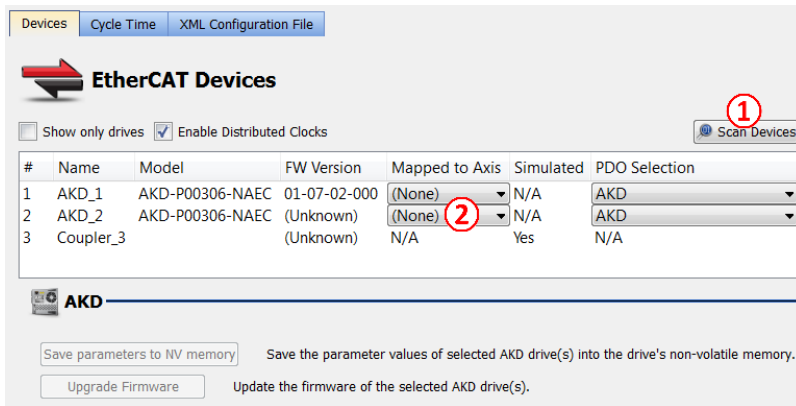
All drives, including the drive in the PDMM itself and the Remote I/O, can be configured by the IDE. In order to connect to an EtherCAT device you must first compile your project. After the project is compiled you must connect to the target:



1. Compile Project
2. Connect to Target

Figure 4-6: Compile and Connect

Next click on the EtherCAT item in the project view to bring up the EtherCAT Devices screen in the IDE workspace. Click on Scan Devices and the IDE will launch a view which automatically identifies your connected device(s). Click OK, to add these devices to the project.



1. Go to the EtherCAT Devices view and press Scan Devices
2. Mapped to Axis in the Application

Figure 4-7: Scan Devices & Mapped to Axis

The devices are added to the project view:

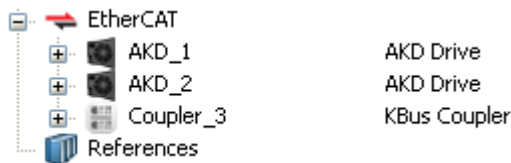


Figure 4-8: Devices Added to Project

To communicate directly with a drive without running a project, first compile and download the project:

Step 1: Compile Project

Step 2: Download Project



1. Compile Project
2. Download Project

Figure 4-9: Compile and Download

Online Configuration Mode must also be active. Click on the EtherCAT item in the project tree and then click on the Online Configuration Mode button next to Scan Devices:

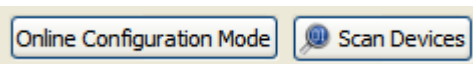


Figure 4-10: Online Configuration Mode

Online Configuration Mode is now active. To configure a particular item in the EtherCAT network, click on the item in the project tree. A set of screens will open in the workspace that enable the user to completely configure the selected drive:

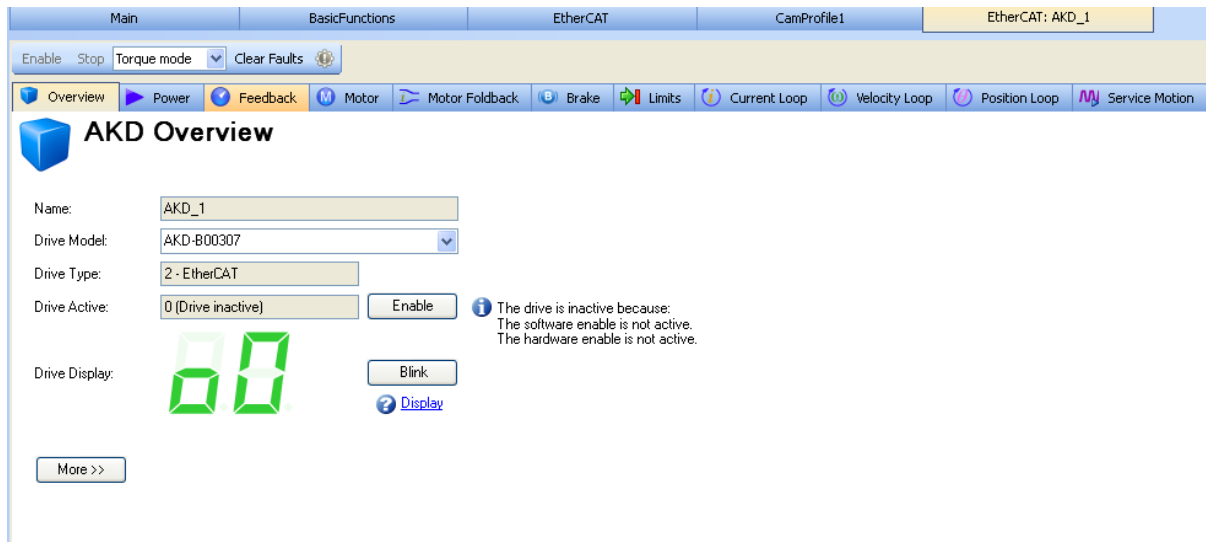


Figure 4-11: AKD PDMM GUI screens

Additionally, the setup wizard will guide you through a set of configuration steps:

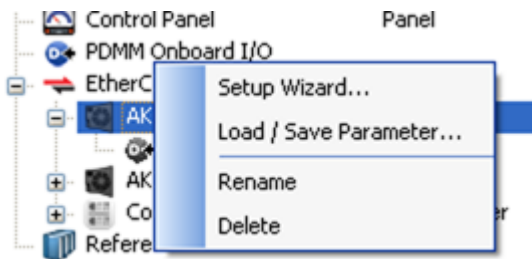


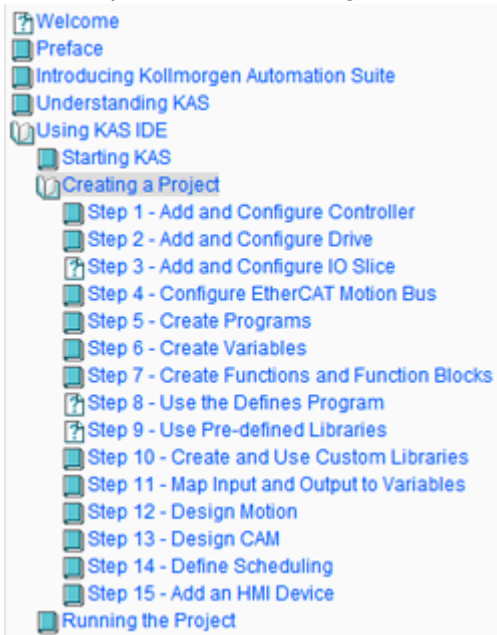
Figure 4-12: Setup Wizard

4.4 Where To Go From Here

Depending upon your need, you should either continue using this manual or use the KAS IDE web help.

- For more information about the KAS IDE embedded WorkBench views, see chapters Using Embedded Workbench Views through Using Parameters and the Terminal Screen (→ p. 168).

- For more information about programming your application and system, see the KAS IDE Online Help. The best place to start is *Using KAS IDE > Creating a Project > Steps 1 - 15*.



5 Initial Drive Setup

| | |
|---|-----------|
| 5.1 Initial Drive Setup | 34 |
| 5.2 AKD PDMM Display Codes | 35 |
| 5.3 AKD PDMM Setup Wizard | 36 |
| 5.4 Settings | 37 |

5.1 Initial Drive Setup

The [AKD PDMM Quick Start Guide](#) provides details for initial drive setup. Initial drive setup consists of the following general steps:

Hardware Installation:

1. Install the drive on your conductive panel and connect the Protective Earth ground.
2. Connect the logic power you will need to operate all of the control logic to X1.
3. Connect the motor power to X2.
4. Connect the feedback to X10.
5. Connect the inputs and outputs you will be using on X7 and X8.
6. Bring AC power to the unit and connect AC power to X3 or X4.
7. Connect drive communications to X11.
8. Confirm that you can communicate with the drive and that your PC is linked to the AKD PDMM.

Software Installation and Drive Communication Setup:

1. Install and start the interface software (KAS IDE).
2. Set the drive IP address using the S1 and S2 switches.
3. Configure the drive using the **Setup Wizard**.

System Requirements

The detailed requirements for each release are indicated in the release notes. In the table below, we list the different versions in which the minimum requirements have changed.

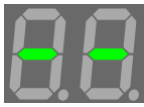



| WB Version | OS | .NET Framework | Comment |
|------------|---|----------------|---|
| 1.14.5.x | Windows 7 (x86/x64) Windows 8 (x86/x64) Windows 10 | 4.6.1 | XP and Vista compatibility is dropped (the application will not install at all as the .NET framework is not supported by these OS). If you need to run WB on one of these OS, please use the 1.14.0.x release . |
| 1.14.0.x | Windows XP Windows Vista Windows 7 (x86/x64) Windows 8 (x86/x64) Windows 10 (x86/x64) | 4.0 | - |

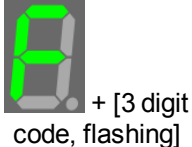
| WB Version | OS | .NET Framework | Comment |
|------------|---|----------------|---------|
| 1.8.0.x | Windows XP Windows Vista Windows 7 (x86/x64) Windows 8 (x86/x64) | 4.0 | - |
| 1.5.0.x | Windows XP Windows Vista Windows 7 (x86/x64) | 4.0 | - |
| 1.4.0.x | Windows XP Windows Vista Windows 7 (x86/x64) | 2.0 | - |

5.2 AKD PDMM Display Codes

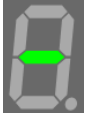
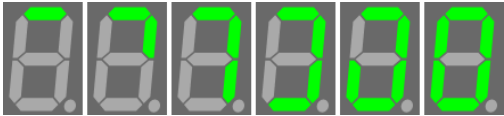
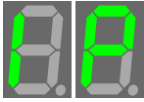


During drive operation the drive display shows the following codes depending on the drive status.

Left Display (Drive LED):

| Display Code | Status |
|---|---|
|  | Powered on and FPGA loading. If the display is locked on this code then operational and resident FPGA images are corrupt. |
|  | Resident loading. |
|  | Resident completed and waiting for the operational to start. |
|  | Normal operation, position mode, no faults. |

| Display Code | Status |
|---|--|
|  + [3 digit code, flashing] | Fault (see Fault and Warning Messages). |

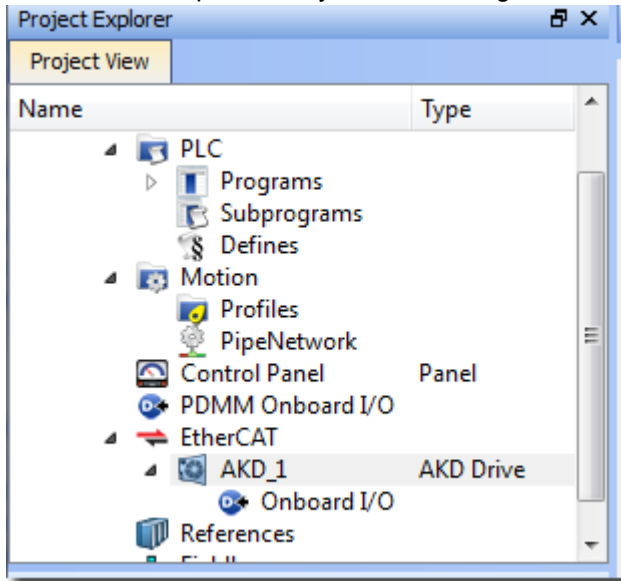
Right Display (Controller LED):

| Display Code | Status |
|---|---|
|  | Powered on and FPGA loading. If the display is locked on this code then operational and resident FPGA images are corrupt. |
|  | Boot sequence. |
|  | Resident completed and waiting for the operational to start. |
|  | Operational - program not running. |
|  | Fault (see Fault and Warning Messages). |

5.3 AKD PDMM Setup Wizard

The Setup Wizard contains step-by-step instructions for configuring a drive for the first time and generating a simple test motion. You can access the Setup Wizard from the Drive Overview screen, or by right clicking on the drive name.

1. Access the Setup Wizard by double-clicking on the drive name in the project view.

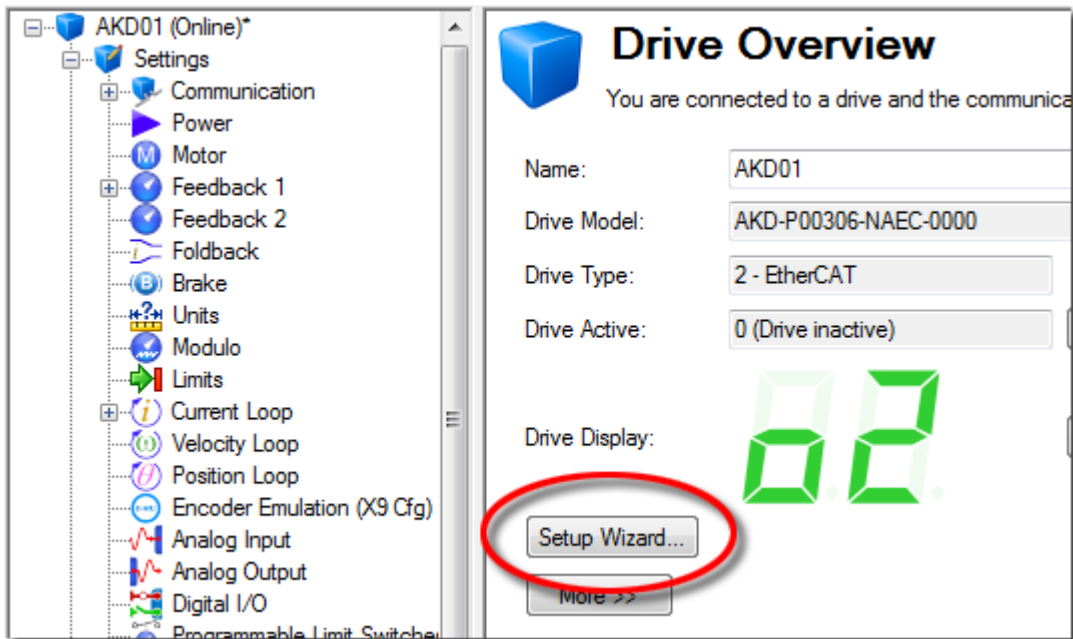


2. Enable Online Configuration mode from the toolbar, by pressing the "Toggle Online Configuration



Mode" button.

3. After the drive is connected, you can press the Setup Wizard button in the Drive Overview screen



The Setup Wizard is useful during the initial setup. The wizard confirms your connection with the drive and leads you through a series of steps to quickly get your drive up and running. With plug and play feedback devices, several steps are skipped (feedback, brake) because the drive automatically configures these settings. For all systems, you can select the units you want to use, configure your operation mode, tune the system, and perform some simple jog moves within the wizard. After you are comfortable with the basic system setup, you can save your settings to the drive and exit the wizard.

5.4 Settings

5.4.1 Navigation Tree

The Settings view allows you to configure the drive to fit your specific application. When you click Settings in the navigation pane located to the left of the EWV¹ screen, additional views for configuring your drive appear. By default, only settings applicable to your current drive operation mode and command source will appear in this tree. To show all settings available for the AKD PDMM (even if these settings are not used with the current operation mode and command source), right click on **Settings** and select **Show all Settings**.

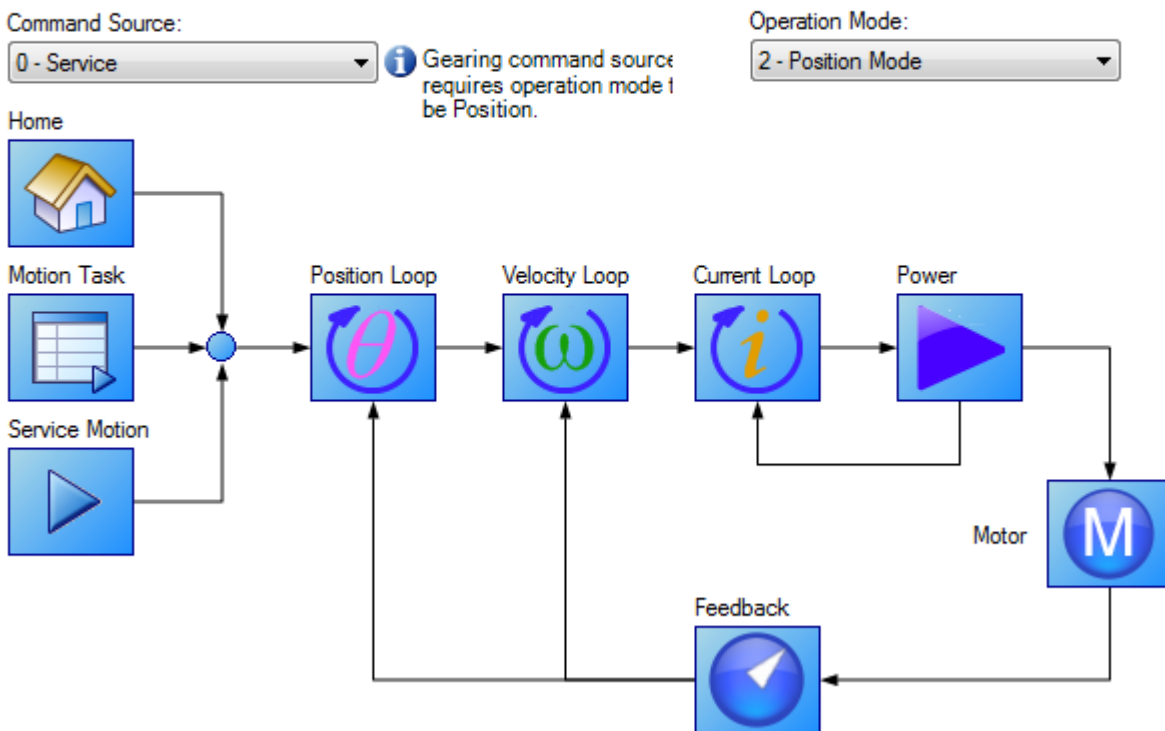
5.4.2 Settings View

From the main Settings view, you can configure the following drive settings:



Settings

Select which mode of operation and command source the drive should work in.



| Button or Dialog Box | Description |
|----------------------|--|
| Command Source | Selects where the command is being provided: 0-Service: You communicate with the drive using the TCP/IP service channel. 1-Fieldbus: The drive is being controlled by commands coming over the fieldbus. 2-Gearing: The position is proportional to the secondary feedback. 3-Analog: The analog input provides a current, velocity, or position command. 1-Fieldbus: For normal KAS Runtime operation 0-Service : For tuning and commanding motion via the AKD PDMM setup screens |

¹Embedded Workbench Views

| Button or Dialog Box | Description |
|----------------------------------|--|
| Operation Mode | <p>Selects the control loop being commanded from the source:</p> <p>0-Torque Mode: Drive controls are based on the current passing through the motor. For a rotary motor, this value is proportional to the torque of the motor; for linear motors, this value is proportional to the force the motor generates.</p> <p>1-Velocity Mode: Drive controls are based on the velocity of the motor.</p> <p>2-Position Mode: Drive controls are based on the position of the motor.</p> |
| Operation Mode | <p>Selects the servo loops that are used in the drive.</p> <p>Mode 0: Current Loop (used primarily during drive setup)</p> <p>Mode 1: Velocity and Current Loops (used primarily during drive setup)</p> <p>Mode 2: Position, Velocity, and Current Loops (Default KAS Runtime mode)</p> |
| Motion and Control Loop Graphics | <p>Allows you to select the details related to each specific loop from a graphical interface.</p> |

Related Topics

Using Command Source and Operating Modes for details on configuring the drive for your application.

6 Configuring Drive Power

| | |
|--|-----------|
| 6.1 Power | 41 |
| 6.2 Operating Voltage | 41 |
| 6.3 Direct DC Mains Operation | 41 |
| 6.4 Regeneration | 43 |

6.1 Power

6.1.1 Drive Setup for Power and Bus

The **Power** screen allows you to confirm Power Bus settings. Review the data on the screen to be certain the bus voltage is at the appropriate levels you expect (approximate input line AC voltage * 1.4). The other values are the appropriate limits for over voltage and under voltage for the particular drive. You can select the under voltage fault mode to trigger either only when the drive is enabled or always.

External Regeneration (→ p. 43) parameters can be set from the Regen screen.

From the **Power** screen, you can view and configure the drive power settings as follows:

| Button or Dialog Box | Description | Parameter |
|---------------------------|--------------------------------------|---------------------------|
| Measured Bus Voltage | Reads the current DC bus voltage. | VBUS.VALUE (→ p. 718) |
| Over Voltage Fault Level | Reads the over voltage fault level. | VBUS.OVFTHRESH (→ p. 712) |
| Under Voltage Fault Level | Reads the under voltage fault level. | VBUS.UVFTHRESH (→ p. 715) |
| Operating Voltage | Sets the operating voltage. | VBUS.HALFVOLT (→ p. 711) |
| Under Voltage Fault Mode | Sets under voltage mode. | VBUS.UVMODE (→ p. 716) |

See Regeneration (→ p. 43) for more information about regen resistors and sizing regen resistors.

6.2 Operating Voltage

Operating voltage can be selected by the user to allow AKD PDMM-xxx07 (480VAC) drives to work on 240VAC input supplies.

The VBUS.HALFVOLT parameter has an effect on the following voltage-thresholds:

- DC-bus over-voltage threshold (see VBUS.OVFTHRESH (→ p. 712)).
- The regen-resistor enable/disable voltage thresholds.
- The inrush-relay enable/disable voltage thresholds.

A power-cycle is needed after changing the value and saving the parameter on the NV memory of the Drive, since the voltage thresholds mentioned above are read during the boot-sequence of the Drive.

6.3 Direct DC Mains Operation

Direct DC input is available on all standard AKD PDMM models. The DC input should be run into the AC input connection. Positive and negative DC lines should use L1 and L2 connections (polarity is not critical). L1 and L2 connections are found on either the X3 connector or the X4 connector depending on the model.

(see Mains supply connection (X3, X4) (→ p. 219) for more information on this connection).

The nominal level of DC voltage applied must be compatible with the voltage fault levels in the drive. You must also consider voltage variations in the DC power supply above and below the nominal value so that nuisance faults are avoided.

When you determine the maximum nominal DC voltage applied to the drive, you should also consider the regeneration circuit, in addition to the over voltage level. Running the drive slightly below the over voltage level is not possible because the drive does not have the capability to dissipate regenerated energy. This practice can also be harmful to the regen circuit. A good practice is not to exceed the nominal DC voltage produced by a standard AC installation.

For the AKD-Mzzzz06, 340 VDC is the equivalent DC voltage for a 240 VAC supply and for the AKD-Mxxxx07, 680 VDC is the equivalent DC voltage for a 480 VAC supply.

The voltage fault levels are also shown in the **Power** screen and depend on the voltage level of drive used.

Voltage ranges are as follows:

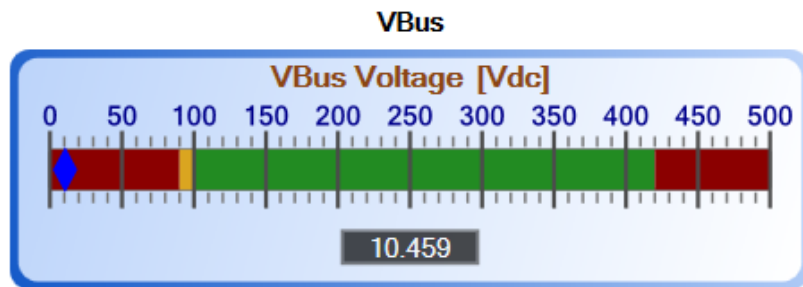
| Model | Under Voltage Level | Over Voltage Level |
|------------|---------------------|--------------------|
| AKD-Mzzz06 | 90 VDC | 420 VDC |
| AKD-Mzzz07 | 380 VDC | 840 VDC |

You can view bus voltage values in the **Power** screen as shown below:



Power

Monitor and configure the different power values.



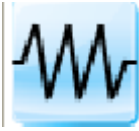
Voltage

Over Voltage Fault Level: Vdc

Under Voltage Fault Level: Vdc

Operating Voltage:

Under Voltage Fault Mode:



6.4 Regeneration

6.4.1 Overview

Regeneration, or "shunting", dissipates energy from the DC bus during deceleration of the motor load. During deceleration, the motor acts as a generator that pumps energy back into the system. If this energy is not dissipated, then the bus energy level can exceed acceptable levels (VBUS.OVFTHRESH (→ p. 712)). If the system exceeds the maximum bus voltage, then the drive generates an over voltage fault (F501n501) and shuts down. A regeneration resistor, or regen resistor, is an external device that dissipates excess energy and allows the drive to function normally during deceleration.

Follow these steps to determine the regen needs for your system and to configure the drive for regen:

1. Calculate motor peak and continuous regenerative energy and use this value to size the regen resistor.
2. Select a compatible regen resistor.
3. Configure regen parameter values in EWV¹.

From the Regen view, the following drive regen settings can be viewed and configured.

| Button or Dialog Box | Description | Parameter |
|-----------------------------|---|--------------------------|
| Regen Resistor Type | Sets the regen resistor type to either -1-External Regen or 0-Internal Regen (if available) . | REGEN.TYPE (→ p. 683) |
| External Regen Resistance | Sets the external, user-defined regen resistor resistance (only visible for external regen). | REGEN.REXT (→ p. 681) |
| External Regen Heat Up Time | Sets the external regen resistor thermal protection time constant (only visible for external regen). | REGEN.TEXT (→ p. 682) |
| External Regen Power | Sets the regen resistor's power fault level for an external regen resistor (only visible for external regen). | REGEN.WATTEXT (→ p. 684) |
| Regen Power | Reads the regen power (only visible for external regen). | REGEN.POWER (→ p. 679) |

6.4.2 Regen Resistor Options

Select "-1 External Regen" from the **Regen. Resistor Type** box (REGEN.TYPE (→ p. 683)) to view the **Select Resistor** drop-down box.

Regen
Configure the regeneration resistor.

Regen. Resistor Type:

External Regen Resistance: Ohm

External Regen Heat Up Time: s

External Regen Power: W

Regen. Power: W

Select Resistor:

| | |
|----------------|-----------|
| BAFP-100-33 | DE-201437 |
| BAFP-200-33 | DE-201438 |
| BAR-250-33 | DE-106254 |
| BAR-500-33 | DE-106255 |
| BAR-1500-33 | DE-106258 |
| BAS-3000-33 | DE-201407 |
| <User Defined> | |

¹Embedded Workbench Views

The **Select resistor** drop-down contains pre-sized resistors for your drive. Select the model regen resistor that you are using and the drive will populate the remainder of the fields. If you are using a nonstandard resistor, then choose **<User defined>** and fill in the appropriate values for your resistor.

NOTE

If you use a nonstandard resistor, contact Kollmorgen technical support to confirm that the nonstandard resistor will work correctly with your system.

6.4.3 Calculating Motor Peak Energy and Regen Resistor Size

In order to determine whether or not your system requires a regen resistor, you must calculate the peak kinetic energy that the motor generates during deceleration and the continuous regeneration energy created by the motor. If this energy exceeds the drive capacity, then you need a regen resistor. In many cases, peak or continuous regenerated energy does not exceed the drive capacity and no regen resistor is needed.

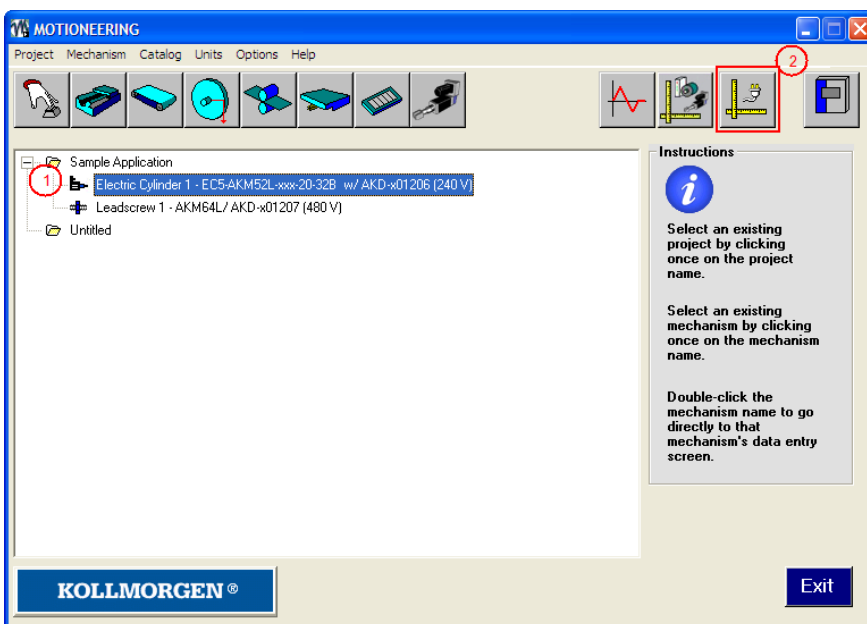
The calculation for peak kinetic energy requires values for several factors that affect the generation of energy in a motion system:

- Load inertia
- Motor inertia
- Motor speed from which deceleration occurs
- Time required to decelerate

Regen resistor sizing information for your particular application can be calculated using the Motioneering® Application Engine. You can download this program here:

http://www.kollmorgen.com/website/com/eng/support/design_tools/motioneering.php

After you install and set up this program, highlight your application (1) and then click on the Power Supply Sizing icon (2).



The application then displays the regen sizing tool; see the application help for further sizing assistance.

Power supply sizing

Project: Sample Application List drives of type: AKD

| Mechanism | Selected system | % rated power | watts |
|-------------|----------------------------|---------------|--------|
| Leadscrew 1 | AKM64L7 AKD-x01207 (480 V) | 57.8 | 3050.8 |

Bus power

Determine by:

- Continuous mechanism load
- Mechanism with motion profile
- Estimated machine tool duty
- Maximum rated system power
- Custom

Power requirements

3050.8 Bus watts

Regeneration requirements

- 178.0 Regeneration watts (avg.)
- 198.5 Maximum resistance (ohms)
- 2796.5 Regeneration watts (peak)

Regeneration

Determine by:

- Simultaneous peak deceleration
- % Duty cycle
- Motion profile

Instructions

Select one or more systems by holding down control key for combined requirements. Select bus power and regeneration methods, and results will be automatically calculated.

NOTE

Multiple servo amplifiers can be coupled via the DC bus connection and thus share the combined regeneration power.

Exit

6.4.4 Selecting a Compatible Regen Resistor

After you calculate the appropriate resistor size, compare the results with the capabilities of the drive and, if necessary, select an external regeneration resistor which matches these capabilities from the chart below. The resistors shown below are included in the KAS IDE setup. If you do not find a match for your application, please contact the Kollmorgen customer support team for further assistance.

| NA Part Number | EU Part Number | Resistor Type | AKD-M00306 | AKD-M00606 | AKD-M01206 | AKD-M02406 | AKD-M00307 | AKD-M00607 | AKD-M01207 | AKD-M02407 |
|----------------|----------------|------------------------------------|------------|------------|------------|------------|------------|------------|------------|------------|
| BAFP-100-33 | DE-201437 | External Resistor, 100 W, 33 ohms | x | x | | | x | x | x | |
| BAFP-200-33 | DE-201438 | External Resistor, 200 W, 33 ohms | x | x | | | x | x | x | |
| BAR-250-33 | DE-106254 | External Resistor, 250 W, 33 ohms | x | x | | | x | x | x | |
| BAR-500-33 | DE-106255 | External Resistor, 500 W, 33 ohms | x | x | | | x | x | x | |
| BAR-1500-33 | DE-106258 | External Resistor, 1500 W, 33 ohms | x | x | | | x | x | x | |
| BAS-3000-33 | DE-201407 | External Resistor, 3000 W, 33 ohms | x | x | | | x | x | x | |
| BAR-600-23 | DE-200613 | External Resistor, 600 W, 23 ohms | | | | | | | | x |
| BAR-1000-23 | DE-200614 | External Resistor, 1000 W, 23 ohms | | | | | | | | x |
| BAS-2000-23 | DE-200615 | External Resistor, 2000 W, 23 ohms | | | | | | | | x |
| BAS-3000-23 | DE-200616 | External Resistor, 3000 W, 23 ohms | | | | | | | | x |
| BAS-4000-23 | DE-200617 | External Resistor, 4000 W, 23 ohms | | | | | | | | x |

| NA Part Number | EU Part Number | Resistor Type | AKD-M00306 | AKD-M00606 | AKD-M01206 | AKD-M02406 | AKD-M00307 | AKD-M00607 | AKD-M01207 | AKD-M02407 |
|----------------|----------------|------------------------------------|------------|------------|------------|------------|------------|------------|------------|------------|
| BAR-500-15 | DE-201439 | External Resistor, 500 W, 15 ohms | | | x | x | | | | |
| BAR-1000-15 | DE-201440 | External Resistor, 1000 W, 15 ohms | | | x | x | | | | |
| BAS-2000-15 | DE-103871 | External Resistor, 2000 W, 15 ohms | | | x | x | | | | |
| BAS-3000-15 | DE-103872 | External Resistor, 3000 W, 15 ohms | | | x | x | | | | |
| BAS-6000-15 | DE-103873 | External Resistor, 6000 W, 15 ohms | | | x | x | | | | |

6.4.5 Configuring Regen Parameter Values

If you use an external regen resistor, then you must configure three regen parameter values. These values are automatically inserted in KAS IDE if the resistor is pre-sized. The chart below summarizes these three additional values:

- **External Regen Resistance** (REGEN.REXT, ohms)
- **External Regen Heat Up Time** (REGEN.TEXT, seconds)
- **External Regen Power** (REGEN.WATTEXT, watts)

These parameters allow the regen resistor to function properly and dissipate power appropriately for your system.

| NA Part Number | EU Part Number | Resistor Type (all resistors UL recognized) | Resistance ohms (REGEN.REXT) | Heat Up Time s (REGEN.TEXT) | Power Rating W (REGEN.WATTEXT) |
|----------------|----------------|--|------------------------------------|-----------------------------------|--------------------------------------|
| BAFP-100-33 | DE-201437 | External Resistor, 100 W, 33 ohms | 33 | 16.5 | 100 |
| BAFP-200-33 | DE-201438 | External Resistor, 200 W, 33 ohms | 33 | 27.5 | 200 |
| BAR-250-33 | DE-106254 | External Resistor, 250 W, 33 ohms | 33 | 22.0 | 250 |
| BAR-500-33 | DE-106255 | External Resistor, 500 W, 33 ohms | 33 | 33.0 | 500 |
| BAR-1500-33 | DE-106258 | External Resistor, 1500 W, 33 ohms | 33 | 25.7 | 1,500 |
| BAS-3000-33 | DE-201407 | External Resistor, 3000 W, 33 ohms | 33 | 77.0 | 3,000 |

| NA Part Number | EU Part Number | Resistor Type (all resistors UL recognized) | Resistance ohms (REGEN.REXT) | Heat Up Time s (REGEN.TEXT) | Power Rating W (REGEN.WATTEXT) |
|----------------|----------------|--|------------------------------------|-----------------------------------|--------------------------------------|
| BAR-600-23 | DE-200613 | External Resistor, 600 W, 23 ohms | 23 | 27.5 | 600 |
| BAR-1000-23 | DE-200614 | External Resistor, 1000 W, 23 ohms | 23 | 27.5 | 1,000 |
| BAS-2000-23 | DE-200615 | External Resistor, 2000 W, 23 ohms | 23 | 77.0 | 2,000 |
| BAS-3000-23 | DE-200616 | External Resistor, 3000 W, 23 ohms | 23 | 84.3 | 3,000 |
| BAS-4000-23 | DE-200617 | External Resistor, 4000 W, 23 ohms | 23 | 77.0 | 4,000 |
| BAR-500-15 | DE-201439 | External Resistor, 500 W, 15 ohms | 15 | 33.0 | 500 |
| BAR-1000-15 | DE-201440 | External Resistor, 1000 W, 15 ohms | 15 | 27.5 | 1,000 |
| BAS-2000-15 | DE-103871 | External Resistor, 2000 W, 15 ohms | 15 | 77.0 | 2,000 |
| BAS-3000-15 | DE-103872 | External Resistor, 3000 W, 15 ohms | 15 | 84.3 | 3,000 |
| BAS-6000-15 | DE-103873 | External Resistor, 6000 W, 15 ohms | 15 | 91.7 | 6,000 |

Related Parameters

REGEN Parameters (→ p. 678)

VBUS.OVWTHRESH (→ p. 713)

VBUS.VALUE (→ p. 718)

7 Configuring Motor Settings

| | |
|---|-----------|
| 7.1 Motor | 50 |
| 7.2 Feedback 1 | 54 |
| 7.3 Extended Multiturn | 57 |
| 7.4 Non-Plug and Play Feedback Devices | 58 |
| 7.5 Foldback | 59 |
| 7.6 Motor I2t algorithm | 62 |
| 7.7 Brake | 64 |
| 7.8 Using Position Capture | 64 |

7.1 Motor

7.1.1 Overview

The **Motor** screen is used to set up or confirm the parameters of the motor that is connected to the drive. In certain cases, based on the feedback type, the motor parameters will automatically be set. The drive will auto-detect feedback devices that are preset with the appropriate feedback and motor parameters when MOTOR.AUTOSET is set to 1 (default). The values the drive uses for commutation and current and velocity loop gains will be populated automatically.

If your motor does not have a plug and play feedback device, then you must turn off the motor autosect feature as shown in the screenshot below (MOTOR.AUTOSET = 0) and select the appropriate motor from the motor parameter database.

vice ▾ | 0 - Torque Mode ▾ | Disable & Clear Faults | Save To Drive | Connect

M **Motor**
 These parameters describe the motor attached to this drive.

Motor Name:

Motor Type:

Motor Autosect:

Peak Current: Ams

Continuous Current: Ams

All of the appropriate Kollmorgen motors compatible with the AKD PDMM drive are contained in the motor database. For motors that are not listed, click **Select Motor** to open a custom motor view in which you can input the appropriate motor parameters.

7.1.2 Motor Setup

When motor autosect is on (MOTOR.AUTOSET = 1), the AKD PDMM will automatically configure motor parameters from the data stored in a supported feedback device (Kollmorgen motors with SFD, Endat, BiSS, Hiperface and Hiperface DSL). If your motor is detected automatically, the parameters in the Motor view are shaded and not accessible. If you have a non-plug and play standard device (such as an incremental encoder or resolver), you can use this screen to enter the standard AKM, Cartridge motor, DDR, or DDL motor. Other motors can be configured from the Selecting a Motor (→ p. 51) screen.

7.1.3 Using the Motor View

The **Motor** view displays parameters related to the specific motor attached to the drive as follows:

- **Motor Name:** The motor part number read from the autosect device, or the name from the motor database. When entering a custom motor name, the motor name should not contain any spaces.
- **Motor Type:** This field allows you to select the proper parameters for a rotary motor or a linear motor (linear motors are a future feature).
- **Field Weakening:** This field allows a surface permanent magnet motor (MOTOR.TYPE = 0) to operate as an interior permanent magnet motor (MOTOR.TYPE = 6). This field can only be set when MOTOR.TYPE = 0.
- **Motor Autosect:** This setting allows the drive to automatically set up a plug and play motor (MOTOR.AUTOSET = 1). With **Motor Autosect** turned off (MOTOR.AUTOSET = 0), you can access the motor database to select a catalog or custom motor.

The next set of parameters displayed are specific to the electrical and mechanical characteristics of the motor connected to the AKD PDMM drive.

- Continuous Current: Motor continuous current rating in Amps rms (MOTOR.ICONT (→ p. 600)).
- Peak Current: Motor peak current rating in Amps rms (MOTOR.IPEAK (→ p. 607)).
- Coil Thermal Constant: Motor coil thermal time constant in mHz (MOTOR.CTF0 (→ p. 598)).
- Inductance (quad, H): Motor rated inductance in milliHenries (MOTOR.LQLL (→ p. 612)).
- Inductance (direct, H): Motor line to line direct axis inductance (MOTOR.LDLL (→ p. 610)). Only visible for MOTOR.TYPE = 6, Rotary Interior Permanent Magnet.
- Inductance Saturation: Motor quadrature axis inductance saturation characteristic (MOTOR.LISAT (→ p. 611)). Only visible for MOTOR.TYPE = 6, Rotary Interior Permanent Magnet.
- Motor Poles: Number of motor poles (MOTOR.POLES (→ p. 618)).
- Motor Phase: Motor phase offset used to set motor commutation as required. For most devices this is set to 0 (MOTOR.PHASE (→ p. 614)).
- Inertia: Motor rotor inertia in Kg-cm² (MOTOR.INERTIA (→ p. 606)).
- Torque Constant: Motor torque constant in Nm/Arms (MOTOR.KT (→ p. 609)).
- EMF Constant: Defines how much voltage is generated at the motors coils (MOTOR.KE (→ p. 608)).
- Motor Resistance: Motor winding resistance in Ohms (MOTOR.R (→ p. 619)).
- Maximum Voltage: Motor maximum rated voltage in Vrms (MOTOR.VOLTMAX (→ p. 632)).
- Maximum Speed: Motor maximum rated speed (MOTOR.VMAX (→ p. 631)).

7.1.4 Selecting a Motor

The **Select Motor** button opens a screen in which you can configure a non-plug and play motor or custom motor.

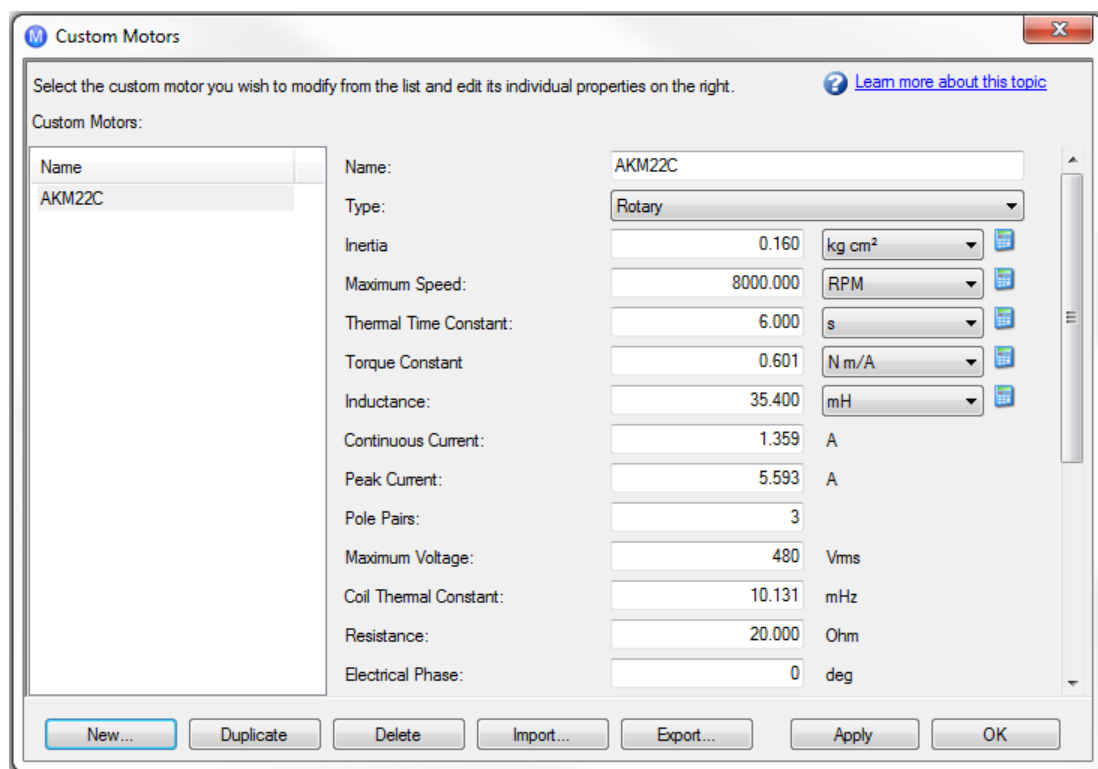
When this screen is opened, KAS IDE displays by default the motor matched to the current motor name attached to the drive. KAS IDE searches matching motor as follows:

1. KAS IDE first checks the motor name with custom motors for a match.
2. If a match is not found, then KAS IDE checks the name with the standard motors database for a match.
3. If a match is not found, then an AKM motor is selected.

For non-plug and play motors, a database of catalog motors is available based on the different Kollmorgen motor families. When you select a motor family, a part number is displayed according to the selected motor family. You can change the part number as needed, and the complete motor name will be displayed according to your selection. This complete motor name is sent to drive. The portions of the part number labeled in bold are required values.

7.1.5 Configuring Custom Motors

From the Select Motor screen, click on **Custom Motor** to create and edit custom motors in the following screen:



In this view, you can import or export a motor parameter file, or create a custom motor of your choice. The appropriate parameters must be chosen as listed. Several of the parameters allow you to select an alternate unit of measure. When building a custom motor file, do not use blank spaces in the name you choose. Once you have configured one or more custom motors, if you select a custom motor from the list and click **OK**, then the selected custom motor will be displayed in the **Select Motor** screen.

The actions available in the custom motor screen include:

- **New.** Allows you to start a new custom motor (with default values) or load a catalog motor you may want to modify.
- **Duplicate.** Makes a copy of the highlighted motor in the custom motor list.
- **Delete.** Deletes the highlighted motor in the customer motor list.
- **Import.** Allows you to import a motor file (*.motor) from another location
- **Export.** Allows you to save the highlighted motor file (*.motor) to another location
- **Apply.** Accepts the values you have entered for the specific motor files you are entering.
- **OK.** Returns you to the **Motor Selection** screen.

When entering any of the motor data, be certain the units are correct. The AKD PDMM drive uses the motor parameters to set up the various feedback loops and limits associated with the motor selected.

Note that if you select a custom motor from the list and click **OK**, then that selected custom motor will be displayed in the Select Motor screen.

7.1.5.1 Validating Motor Parameters

When you click **OK** in the **Select Motor** screen, KAS IDE validates the range with the drive. If any error is found, an error screen is displayed. Click **Continue** to set the motor parameters in the drive. Click **Cancel** to close this screen.

If errors occur while setting the motor parameters, an error screen indicates which parameters require additional attention.

7.1.6 Motor Derating

Motor derating occurs while using a motor in the AKM or VLM series. If a brake is selected or a feedback type other than Resolver is selected, the continuous current will be derated.

Resolver with no brake (no derate):

M Motor

These parameters describe the motor attached to this drive.

Motor Name:

Motor Type:

Motor Autoset:

Continuous Current: Ams

Peak Current: Ams

Select Motor

M Motor [Learn more about this topic](#)

Select the motor that is attached to the drive.

To attach a motor, first select Motor family and then Motor series.

Motor Family:

Frame/Winding: Mount: Shaft: Connectors: Brake: Feedback:

To create new or edit existing custom motors:

SFD with no brake (derated):

M Motor
 These parameters describe the motor attached to this drive.

Motor Name:

Motor Type:

Motor Autoset:

Continuous Current: Ams

Peak Current: Ams

Select Motor

M Motor
 Select the motor that is attached to the drive. [Learn more about this topic](#)

To attach a motor, first select Motor family and then Motor series.

Motor Family:

Frame/Winding: Mount: Shaft: Connectors: Brake: Feedback:

To create new or edit existing custom motors:

Feedback dropdown menu:

- 1- 1024 LPR digital encoder w/commutation
- 2- 2048 LPR digital encoder w/commutation
- C- Smart feedback device (SFD)
- R- Resolver
- GC Single-turn capacitive HIPERFACE encoder
- GD Multi-turn capacitive HIPERFACE encoder

7.1.7 Motor Temperature

The motor temperature settings are located in the nested motor temperature view. If your motor has thermistor with linear characteristics (MOTOR.RTYPE = 2, 3, or 4), the Actual Motor Temperature field will display the value of MOTOR.TEMPC. For other values of MOTOR.RTYPE this field will be hidden.

Device Topology

- Start Page
- CN (Online)*
 - Settings
 - Communication
 - Power
 - Regen
 - Motor
 - Motor Temperature
 - Feedback 1
 - Feedback 2
 - Foldback
 - Brake
 - Units
 - Modulo
 - Limits

M Motor Temperature
 Monitor the motor temperature and configure faults/warnings.

Thermal Resistor Type:

Actual Thermistor Value: Ohm

Fault Level: Ohm 0 means no fault

Warning Level: Ohm 0 means no warning

Actual Motor Temperature: degC

Many motors have a thermistor embedded in their windings. The thermistor is usually connected through the feedback connector.

Related Parameters

MOTOR Parameters (→ p. 590)

7.2 Feedback 1

The Feedback view allows you to configure the primary position feedback device fitted to your motor.

When you select your feedback device from the Feedback Selection list, the appropriate feedback configuration choices appear below the dial.

7.2.1 Overview

The AKD PDMM offers a variety of feedback solutions, which allow you to optimize your system based on your specific machine needs. The table below lists the currently supported feedback types. Your motor model number will indicate the type of feedback that you have.

Kollmorgen motors with digital feedback devices (such as SFD, Endat, BiSS and Hiperface, Tamagawa) are plug and play. With these motors, all feedback and motor settings are configured automatically. Third party motors or Kollmorgen motors with non-digital feedback types require that parameters be entered manually. See Non-Plug and Play Feedback Devices (→ p. 58)).

The following table lists current support for primary and secondary feedback:

| | | Primary | Secondary |
|-------------------------------|---------------------------------------|---------|-----------|
| Resolver | Std & Multi pole | Yes | No |
| SFD, SFD3 | | Yes | No |
| Incremental (Digital) Encoder | With Halls and Index | Yes | No |
| | No Halls with Index | Yes | Yes |
| | No Halls with No Index | Yes | Yes |
| Analog Sin/Cos Encoder | With Digital Halls | Yes | No |
| | With Digital Halls and Analog Index | No | No |
| | No Halls and No Index | Yes | No |
| EnDAT 2.1 | Single & Multi Turn | Yes | No |
| EnDAT 2.2 | All Digital | Yes | Yes |
| | Analog / Digital | Yes | No |
| BiSS, BiSS Mode C Renishaw | All Digital (Mode C) | Yes | No |
| | Analog / Digital | Yes | No |
| Hiperface, Hiperface DSL | Analog / Digital; Single & Multi Turn | Yes | No |
| | All Digital | Yes | No |
| Halls Sensor Only | Digital | Yes | No |

7.2.2 Using Feedback Options

Use the **Feedback** screen to set up your system to match the proper feedback device. By default, the drive uses the **Auto** setting to detect feedback devices. This setting allows the drive to test the feedback device to see if it is a recognized plug and play device. If the drive recognizes the device, then all the parameters for that device and motor are loaded into the drive. Both the feedback and the motor information are now present in the drive and the system is operable.

If the feedback is a non- plug and play device, then you can choose from the list of supported devices in **Feedback Selection** list and then enter the line count manually. The following sections describe each supported device available in the **Feedback Selection** list and the input information required to configure each device.

7.2.2.1 Auto

This is the default setting and is used to determine if a plug and play device is available. If a plug and play device is available, the **Auto** mode is replaced by the feedback device detected, along with the appropriate resolution settings.

7.2.2.2 None

This setting can be used if no feedback device is connected to the X10 connector. This may be the case if FB2 or FB3 is used for position or velocity control.

7.2.2.3 Incremental Encoder

The incremental encoder is a non-plug and play device. Incremental encoders are available in a variety of line counts. If you select an incremental encoder option, the encoder resolution must be entered into the **Rotary Encoder Resolution** box. The units for this field are in counts per revolution, which is post-quadrature (multiplies the lines per revolution by 4).

7.2.2.4 Sine Encoder

Sine encoders are offered with different data communication protocols. These include Endat, BiSS, Hiperface, and others. A standard sine-cosine encoder with simple analog communication is not a plug and play device. As with the incremental encoder, the line count is entered in the **Rotary Encoder Resolution** box.

7.2.2.5 Endat 2.1, Endat 2.2

These feedback device are plug and play when ordered as an option with AKM motors.

7.2.2.6 BiSS Analog

These feedback devices are plug and play when ordered as an option with AKM motors.

7.2.2.7 Hiperface, Hiperface DSL

These feedback devices are plug and play when ordered as an option with AKM motors.

7.2.2.8 Resolver

The resolver feedback option is not plug and play. When selecting the resolver option, three specific parameters are set by default for the standard AKM resolver: phase lag, transformation ratio, and feedback poles. Currently, the AKD PDMM does not support non-Kollmorgen standard resolver options.

NOTE

The resolver frequency is set to 6875Hz and cannot be adjusted.

7.2.2.9 SFD, SFD3

Smart Feedback Device (SFD) is Kollmorgen's most popular plug and play device. SFD allows for quick and easy setup from the **Auto** mode, which automatically configures the drive with the motor and feedback parameters.

7.2.2.10 Tamagawa

This feedback type is not available on models with extension type NA (AKD PDMM Models (→ p. 26)).

7.2.2.11 Halls Only

This feedback type only supports operating in Torque mode and Velocity mode.

Due to heavy quantization, large current spikes occur at hall transition points, it is advisable to tune the observer to a low frequency (like 10Hz) and set the observer gain appropriately to achieve the best operating results.

7.2.3 Wake and Shake Overview

Motors which have no way to automatically identify commutation must run Wake and Shake. Motors without Halls fall into this category, as well as motors whose hall effect sensors were manually installed, and not aligned to a motor phase of 0 degrees.

The AKD PDMM will determine if Wake and Shake is needed for safe operation based on feedback type. If needed, the drive will automatically arm Wake and Shake. If Wake and Shake does not complete, a fault will be generated. When that fault is cleared, Wake and Shake will be automatically armed again.

NOTE

If the user attempts to manually disarm Wake and Shake on an AKD PDMM which requires it, a fault will be thrown and Wake and Shake will automatically arm again.

If Wake and Shake is armed on a motor with a feedback type that does not normally require Wake and Shake, the algorithm will still execute. If Wake and Shake fails, a fault will be generated. After the faults are cleared, Wake and Shake will automatically arm again.

NOTE

If Wake and Shake has been manually armed, it can be disarmed by clicking on the "disarm" button on the Wake and Shake screen, or by executing the WS.DISARM (→ p. 761) command over telnet.

The AKD PDMM has multiple modes of completing Wake and Shake, each with different benefits for different operating conditions.

Mode 0:

Mode 0 is the fastest of the Wake and Shake methods; it executes and completes in a few seconds, and is ideal for applications that need the fastest start-up time. The tradeoff for this fast execution time is complexity in configuration. There are a lot of parameters that must be precisely tuned for this mode to operate correctly.

For more information see Using Wake and Shake Mode 0 (WS.MODE 0) (pg 1)

Mode 1:

Mode 1 is a more traditional method of completing Wake and Shake. It is also referred to as Commutation Alignment and Pole Locking. Mode 1 takes a little longer to execute than Mode 0, but is easier to configure. Only WS.IMAX (→ p. 765) (current to be applied) and WS.TSTANDSTILL (time to apply current) need to be configured. The default values work for most motors.

For more information see Using Wake and Shake Mode 1 (WS.MODE 1) (pg 1)

Mode 2:

Mode 2 is the easiest of the Wake and Shake modes. It requires no parameters to be configured (setting WS.FREQ for sine frequency excitation is optional, and the default value of 10,000 Hz will work for most cases). This Auto-Commutation finding algorithm can take as long as 30 seconds to complete. Measurements are taken by exciting the motor with a sine excitation as the motor phase is adjusted. After a series of data points have been taken a DFT analysis is done to determine the motor phase offset for commutation.

7.3 Extended Multiturn

If using a multiturn feedback where the machine period is not an integer divisor of the encoder period, the full turn count must be saved so that the exact position can be recovered after a power cycle. The machine period is the number of units required by the machine to complete one operational cycle. The encoder period is the maximum amount of travel measurable by the encoder. The encoder period is calculated using the following formula:

$$\text{encoder period} = 2(\text{number of encoder turn bits})(\text{units for one turn of an end system})(\text{gear ratio})$$

In some applications, a multiturn feedback is not enough to ensure that the position is known after a power cycle. These applications are those where:

$$\text{Machine period} \neq \text{encoder period} * N$$

In these cases, it is necessary to track and store the number of turn bits beyond the bits stored in the feedback. These bits can be stored by setting FB1.EXTENDEDMULTITURN (→ p. 460) = 1. The supported feedback types include all multiturn encoders: BiSS, EnDAT, Hiperface and Tamagawa (the algorithm requires 12 multiturn bits). If FB1.EXTENDEDMULTITURN is set to 1 on an invalid feedback type then an error will be reported.

7.4 Non-Plug and Play Feedback Devices

To set up a AKD PDMM drive with a non-plug and play feedback device (for example a resolver or an encoder), you must select a motor from the list of standard or custom motors or enter the motor parameters manually. Once the motor data is entered into KAS IDE, an initial set of parameters can be calculated and downloaded to the drive.

7.4.1 Parameters

The following parameters can be initialized to their default values or calculated from user-supplied motor data:

| | | | |
|-------------|-----------------|---------------------------|--------------------------------------|
| IL.KP | IL.LIMITN | VL.KP | PL.KI = 0 (Default Value) |
| IL.KFFACC | IL.LIMITP | VL.KI | PL.INTINMAX = 0.419 (Default Value) |
| IL.KBUSFF | IL.PWMFREQ | VL.LIMITN | PL.INTOUTMAX = 0.419 (Default Value) |
| IL.FRICTION | IL.KVFF | VL.LIMITP | MOTOR.IPEAK |
| IL.OFFSET | IL.FOLDFTHRESHU | VL.THRESH | MOTOR.ICONT |
| IL.INTEN | IL.FOLDWTHRESH | VL.KVFF | MOTOR.PITCH |
| IL.IVFB | IL.MFOLDD | PL.KP | MOTOR.POLES |
| IL.KPDRATIO | IL.MFOLDT | PL.KD = 0 (Default Value) | MOTOR.TYPE |

7.4.2 Calculations

KAS IDE uses the following equations to calculate parameter values.

7.4.2.0.1 Current Loop

The current loop proportional gain (IL.KP) must be such that the current loop closed loop crossover frequency/bandwidth (BW) nominal is the lesser of 2000 Hz or (PWM Frequency/4).

Then, with this frequency:

$$IL.KP = 2 * \pi * (\text{desired bandwidth in Hz}) * (\text{motor L line-line in H})$$

Setting the D and Q components

7.4.2.0.2 Velocity Loop

$$VL.KP = (2 * \pi * 75) * (2 * J_m / K_t) = 300 * \pi * J_m / K_t$$

$$VL.KI = 5$$

7.4.2.0.3 Slider Tuning

The slider tuning algorithm in KAS IDE currently uses the following algorithm.

7.4.2.0.4 Input - Motor Data

The values for inertia, J_m (Kg / cm²), and torque constant, K_t (Nm/A), are obtained from either the SFD or the motor model number that you select.

7.4.2.0.5 Constants

Velocity Loop Bandwidth – BW = The default value is 75Hz.

Input - Inertia Ratio – Q = The default is 1.

7.4.2.0.6 Output - Control Loop Gains

$$VL.KP = 2 * \pi * BW * J_m * (1+Q) * 0.0001/K_t$$

$$VL.KPI = BW * 0.08^{-1} \text{ (minimum of 1)}$$

$$PL.KP = BW/5$$

7.5 Foldback

The foldback feature in the AKD PDMM protects both the motor and the drive from overheating. Two current foldback algorithms run in parallel in the drive: the drive foldback algorithm and the motor foldback algorithm. Each algorithm uses different sets of parameters. Each algorithm has its own foldback current limit, IL.DIFOLD (drive foldback) and IL.MIFOLD (motor foldback). The overall foldback current limit is the minimum of the two at any given moment.

$$IL.IFOLD = \min(IL.DIFOLD, IL.MIFOLD).$$

Foldback is not the same as current limits. Instantaneous current limits for the drive are set by the positive peak current (IL.LIMITP) and negative peak current (IL.LIMITN) in the Limits view in [EWV¹](#). The foldback algorithms may reduce the current output to the motor in spite of the current limit settings.

7.5.1 Drive Foldback

The drive foldback algorithm monitors current feedback; since this is a monitoring function, the drive foldback parameters are not user configurable. If the current feedback exceeds the continuous current rating of the drive (DRV.ICONT), then the algorithm decreases the current to the DRV.ICONT level. For example, under a step command input condition, the foldback algorithm allows maximum peak current (DRV.IPEAK) output from the drive for a short period of time (up to IL.DFOLDD time), after which the drive begins an exponential foldback (with time constant of IL.DFOLDT) of the current to the drive's continuous current.

It takes a few seconds for the exponential decay to drop from the drive's peak current to its continuous level. A recovery time, when the feedback current is below DRV.ICONT level, is required to allow current above DRV.ICONT level again. A recovery time of IL.DFOLDR with 0 current allows the drive to apply DRV.IPEAK current for IL.DFOLDD time.

7.5.2 Setting up motor foldback

Motor foldback is set up automatically when using a plug and play motor or when a particular motor is selected from the EWV database.

If you are using a custom motor, use the Motor view in EWV to set custom values needed for foldback configuration. The parameter entries required for the drive to apply motor foldback protection properly are coil thermal constant (MOTOR.CTFO), peak current of the motor (MOTOR.IPEAK), and continuous current of the motor (MOTOR.ICONT). These values are used to setup the algorithm for motor foldback.

7.5.3 Setting Fault and Warning Levels

The **Motor Current Limit** and **Overall Current Limit** boxes show status variables that are constantly updated by the foldback algorithm. As current is applied above the drive or motor continuous rating, the capacity for the application of peak current starts to decrease. The motor current limit and overall current limit are actively decreased. If the move profile requires less than continuous current rating for a period of time, the

¹Embedded Workbench Views

Motor Current Limit and Overall Current Limit begin to increase until they reach maximum foldback capacity once again.

When Motor Current Limit or Overall Current Limit < Warning Level, an n524 status warning is triggered. When Motor Current Limit or Overall Current Limit < Fault Level, an F524 fault is triggered and the drive power stage is disabled. The load then coasts to a stop.

In the **Foldback** screen, setting the Warning Level to 0 turns off the warning feature. Otherwise, the Warning Level must be set above the Fault Level, but below the Motor Current Limit and Overall Current Limit to trigger a warning.

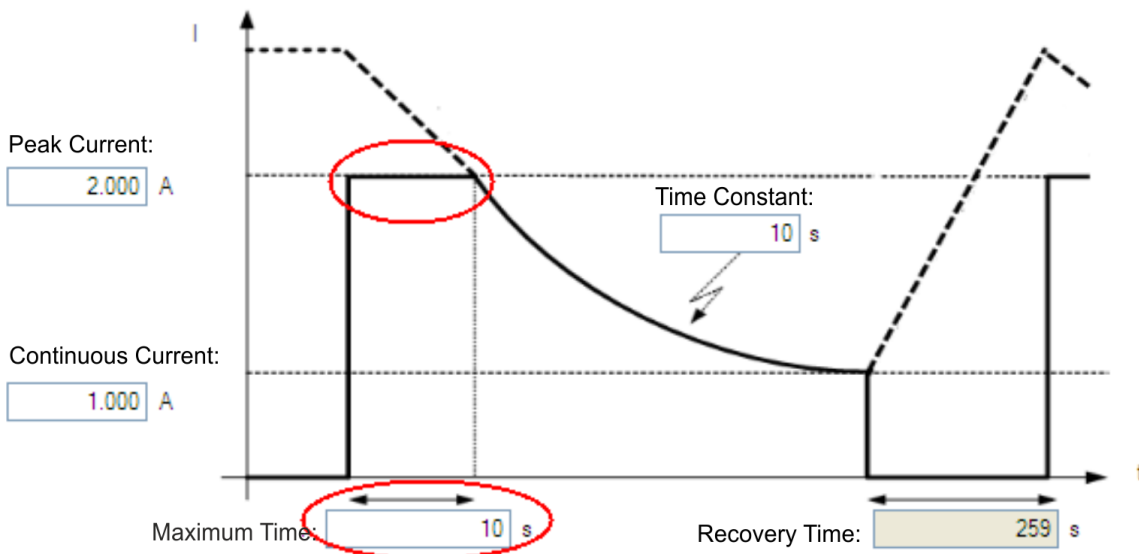
| | | | |
|------------------------|---|-------------------|--|
| Motor Current Limit: | <input type="text" value="16.508"/> Ams | Fault Level: | <input type="text" value="9.000"/> Ams |
| Overall Current Limit: | <input type="text" value="16.508"/> Ams | User Fault Level: | <input type="text" value="9.000"/> Ams |
| Current Command: | <input type="text" value="0.000"/> Ams | Warning Level: | <input type="text" value="0.000"/> Ams |

If User Fault Level is set above Fault Level, the User Fault Level will be ignored. The User Fault Level is used to increase the time the drive is operated in foldback mode without faulting. For instance if the default Fault Level is 9.000 Arms and a User Fault Level is set to 7.5 Arms, the Fault Level is changed to 7.5 Arms. This configuration effectively increases the time that foldback will be applied to the drive before faulting.

| | | | |
|------------------------|---|-------------------|--|
| Motor Current Limit: | <input type="text" value="16.508"/> Ams | Fault Level: | <input type="text" value="7.500"/> Ams |
| Overall Current Limit: | <input type="text" value="16.508"/> Ams | User Fault Level: | <input type="text" value="7.500"/> Ams |
| Current Command: | <input type="text" value="0.000"/> Ams | Warning Level: | <input type="text" value="0.000"/> Ams |

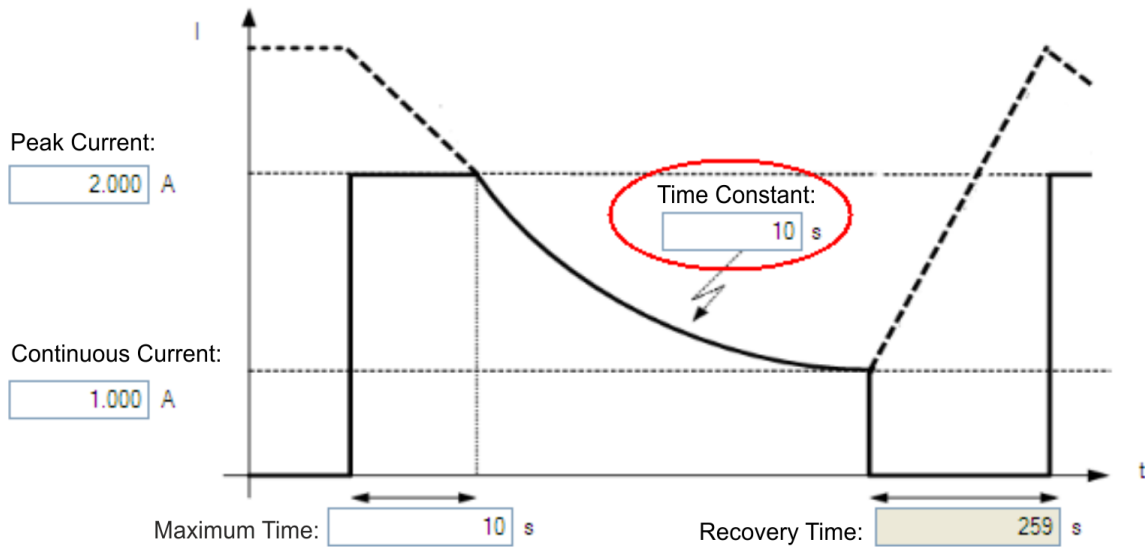
7.5.4 Motor Peak Current Time

Peak current (MOTOR.IPEAK) along with coil thermal constant (MOTOR.CTFO) are used to determine the maximum time the motor can sustain peak current. The maximum time (IL.MFOLDD) is displayed in the **Foldback** screen as shown below:



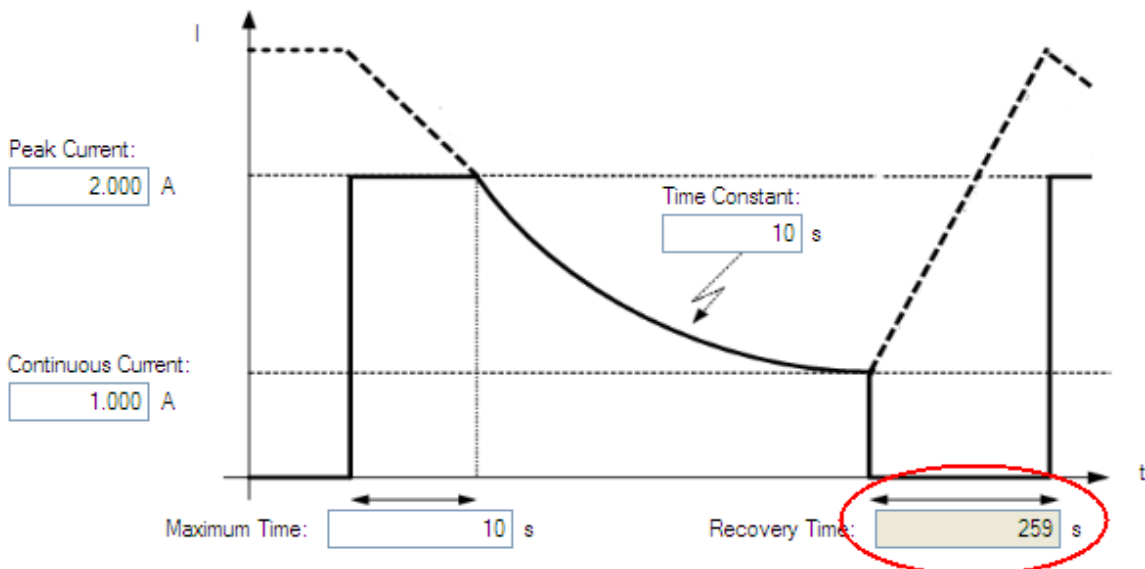
7.5.5 Motor Foldback Ramp

Once the maximum time for motor peak current has elapsed, if the move profile still demands peak current from AKD PDMM, the drive will exponentially lower the current applied to the motor. The Time Constant (IL.MFOLDT) dictates the profile. A smaller time constant represents a steeper decline in current applied to the motor.



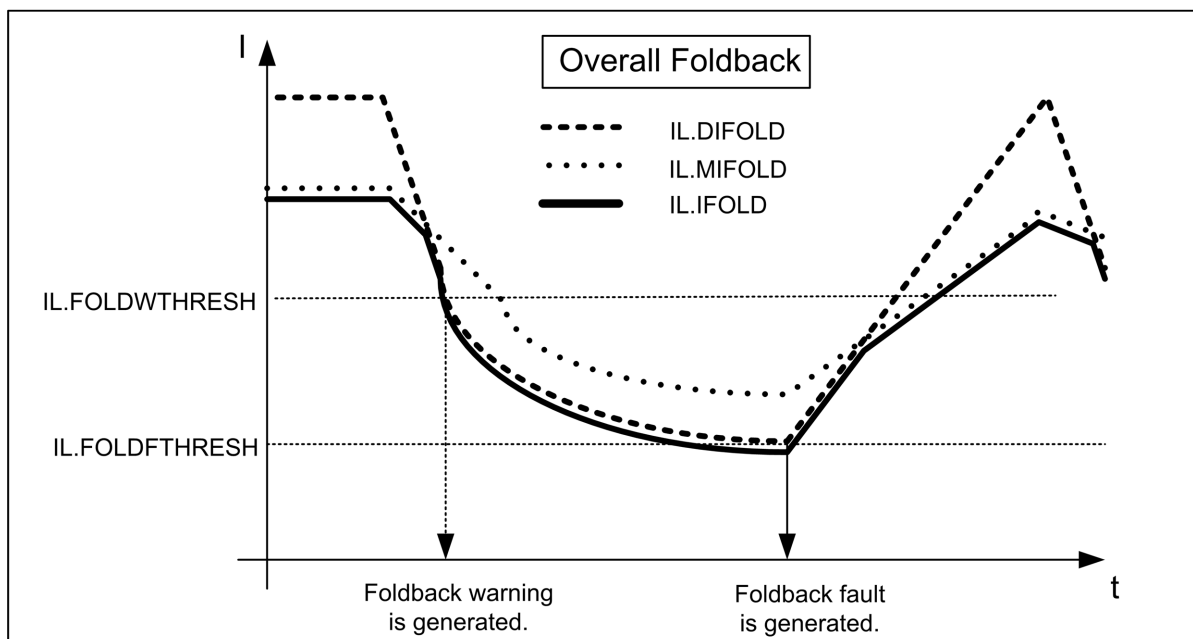
7.5.6 Motor Recovery

Once the peak motor current available has reached the continuous current of the motor, the motor needs Recovery Time (IL.MFOLDR) to cool down. Full Recovery Time (IL.MFOLDR) at 0 current is required for the motor to reach full maximum capacity in the shortest amount of time. The drive can command a current less than continuous current to continue driving the load, but the recovery time for full maximum capacity is increased.



7.5.7 Overall Foldback

The overall limit is the momentary minimum value between the drive foldback and the motor foldback. The overall foldback is shown in the diagram below. You can set the warning and the fault levels as shown in the diagram.



7.6 Motor I²t algorithm

Set IL.MIMODE (→ p. 582) to 1 to use the I²t algorithm for motor protection. If MOTOR.ICONT (→ p. 600) has not been automatically configured, set it to the continuous rating of the motor. Then configure the warning threshold using IL.MI2TWTHRESH (→ p. 580).

7.6.1 Parameters

The following parameters are used for the motor load I²t algorithm.

IL.MI2T (→ p. 579): Current motor load in [%].

IL.MI2TWTHRESH (→ p. 580): Warning threshold in [%]. The AKD PDMM generates a warning n309 in case that the IL.MI2T exceeds this warning threshold.

IL.MIMODE (→ p. 582): Motor protection mode. This parameter determines whether the motor foldback algorithm or the I²t algorithm is responsible for the motor protection.

7.6.2 Functional description of the motor I²t algorithm

The connected motor can be protected using two different kinds of algorithms:

- Motor foldback as described in Foldback (→ p. 59).
- The motor I²t algorithm.

The algorithm used by the AKD PDMM can be chosen using IL.MIMODE.

The motor I²t algorithm calculates continuously an artificial motor load value and limits the supplied current in case that the motor load reaches 100[%].

The motor load is calculated by applying a low pass filter with time constant τ to the I² load.

Increasing motor load:

$$IL.MI2T = 100\% * \sqrt{\frac{i_q^2 + i_d^2}{MOTOR.ICONT^2}} \cdot (1 - e^{-\frac{t}{\tau}})$$

Decreasing motor load:

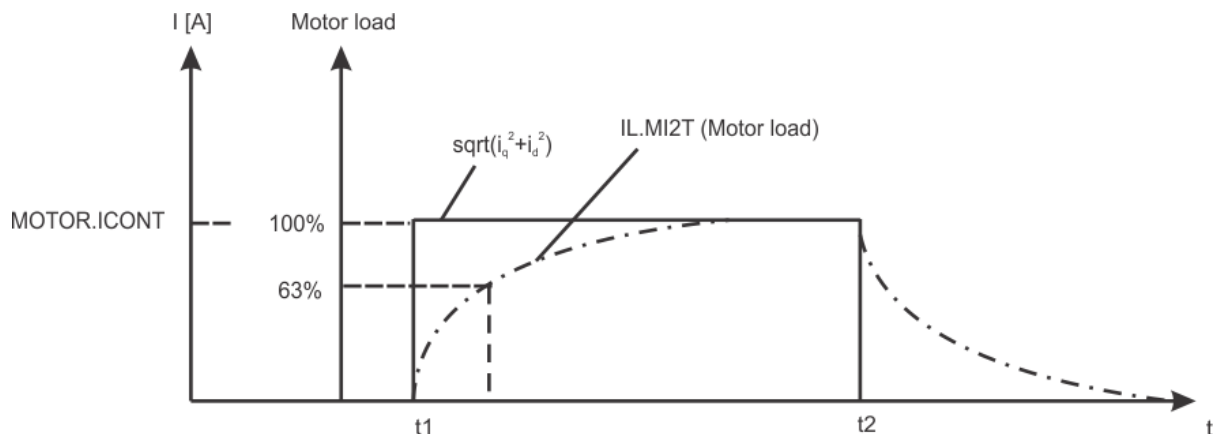
$$IL.MI2T = 100\% * \sqrt{\frac{i_q^2 + i_d^2}{MOTOR.ICONT^2}} \cdot e^{-\frac{t}{\tau}}$$

The variable i_q describes the current generated by the AKD PDMM, which is responsible for achieving torque on the motor shaft whereas i_d describes the generated current, which has an impact on the electrical field of the motor.

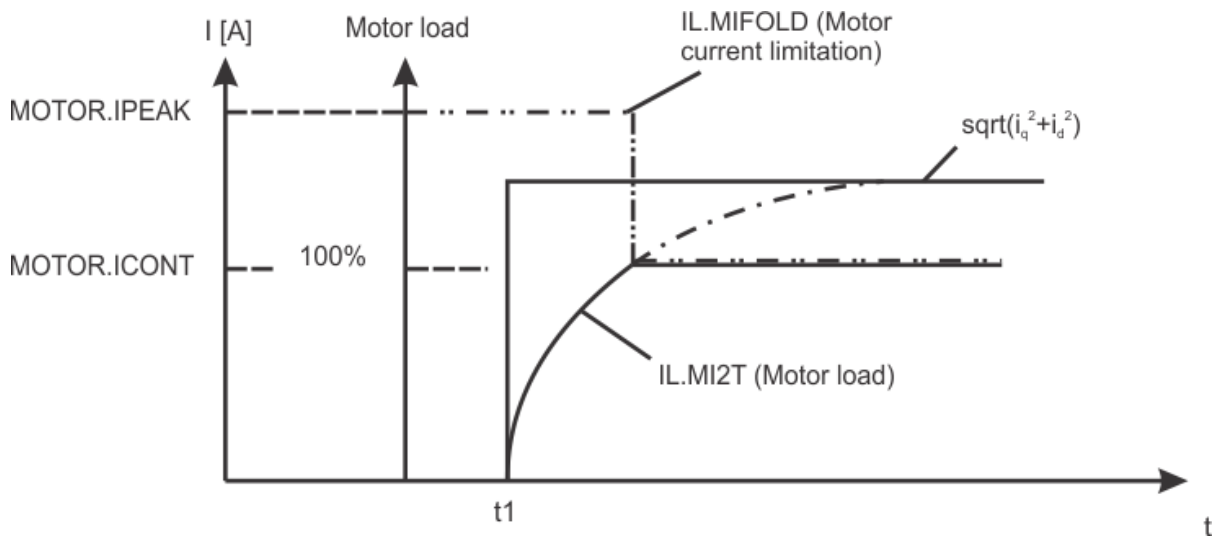
The motor load is described as an exponential function with the time constant τ calculated in the following manner:

$$\tau[s] = \frac{1}{2 \cdot \pi \cdot MOTOR.CTF0[Hz]}$$

The figure below shows the progress of the motor load in case that continuous current is applied to the motor starting from t_1 and in case that no current will be applied any more starting from t_2 . The motor load will reach a 100% value after $5 \cdot \tau$ in case that continuous current is applied by the AKD PDMM.



A current limitation is used by the AKD PDMM in order to protect the motor-coil from overheating. The variable "IL.MIFOLD" (\rightarrow p. 581), which is responsible for limiting the current in the motor windings (see Fold-back (\rightarrow p. 59)) will be set to $MOTOR.ICONT$ (\rightarrow p. 600) in case that $IL.MI2T$ reaches a value of 100%. The value of $IL.MIFOLD$ will be set to $MOTOR.IPEAK$ (\rightarrow p. 607) as soon as the value of $IL.MI2T$ falls under 95%. The following figure describes the limitation of the supplied current.



7.6.3 Details about IL.MI2T

IL.MI2T is calculated based on measured motor currents (IL.FB (→ p. 555) and IL.DFB (→ p. 553)). When the motor is disabled, instead of zero percent in IL.MI2T or 0 IL.FB/IL.DFB, there will be a small percentage number, usually 1-3% depending on the drive. This is caused by current measurement noise/resolution limit and is normal.

7.7 Brake

The brake output on connector X2 controls a mechanical brake that optionally may be fitted to a motor. The brake is applied and released relative to the **Drive Active** state of the drive. You can modify the release and apply delays using the parameters shown below.

| Button or Dialog Box | Description | Parameter |
|----------------------|---|---------------------------------|
| Brake ControlState | Sets whether the motor has a brake. | MOTOR.BRAKE |
| Time to Release | The time, set by the user, required to mechanically disengage the brake. After this time, motion commands are accepted. | MOTOR.TBRAKERLS |
| Time to Apply | The time, set by the user, required to mechanically engage the brake. After this time, the drive will disable. | MOTOR.TBRAKEAPP |

7.8 Using Position Capture

7.8.1 Overview

Position capture allows you to precisely determine what the motor position (or drive clock time) was when a specific event triggers. The AKD PDMM drive uses two independent captures, which operate similarly.

Position capture is used in precision environments, where the motor may be moving at very high velocities, an IO is triggered, and you must know exactly where the motor was when the event occurred. Homing algorithms often use position capture.

Position capture will capture the motor position (or drive clock time) when the capture trigger is activated. The position capture or drive lock time capture will happen in less than 3 microseconds of the input transition. This assumes the use of high speed input 1 or 2 with the filter turned off. The capture engine can be re-armed and ready for another capture in less than 62.6 microseconds. This capture engine allows the position capture to obtain more accurate results than those obtained using the scope or recorder clock.

7.8.2 Configuring Position Capture

To configure the position capture, select **Position Capture** from the **Settings** group:

PC

Position Capture

The drive will be able to capture the position of the axes

Position Capture 0

Capture Mode: 0 - Single-shot Position

Capture FB Source: 3 - Standard position

Arm

Captured Value: 0.000 deg

Capture Parameters

Source: 0 - DIN 1

Edge: 1 - Rising Edge

Pre Condition

Condition: 0 - Trigger edge (ignore precc

Source: 0 - DIN 1

Edge: 1 - Rising Edge

Position Capture 1

Capture Mode: 4 - Auto-armed position

Capture FB Source: 3 - Standard position

Armed

Captured Value: 0.000 deg

Capture Parameters

Source: 0 - DIN 1

Edge: 1 - Rising Edge

Pre Condition

Condition: 0 - Trigger edge (ignore precc

Source: 0 - DIN 1

Edge: 1 - Rising Edge

7.8.2.0.1 Setting the Capture Source (CAP0.FBSOURCE, CAP1.FBSOURCE (pg 1))

CAP0.FBSOURCE and CAP1.FBSOURCE select the source of the captured value. Data for all source values is retrieved with CAP0.PLFB, CAP1.PLFB (pg 1).

| Value | Description |
|-------|--|
| 0 | Captures from feedback 1 (FB1), independent of the value of PL.FBSOURCE. The units are the same as those used for PL.FB. |
| 1 | Captures from feedback 2 (FB2). |
| 2 | Captures from feedback 3 (FB3). |
| 3 | Standard position capture, which stores PL.FB (→ p. 644). The feedback number is determined by PL.FBSOURCE (→ p. 645). |
| 4 | Captures from EXTENCODER.POSITION (AKD BASIC drives only). |

7.8.2.0.2 Setting the Trigger Source (CAP0.TRIGGER, CAP1.TRIGGER (pg 1))

The trigger source determines which input on the drive causes the position capture to trigger.

| Option | Description |
|--------|--|
| 0 - 6 | These options trigger on the Digital Input 1 pin through Digital Input 7 pin, respectively. |
| 7 - 9 | These options trigger on the X9 connector RS485 Input 1 through RS485 Input 3, respectively. |
| 10 | This option triggers on the primary encoder index. |

7.8.2.0.3 Setting the Capture Mode (CAP0.MODE, CAP1.MODE)

The capture mode determines what information is saved on the drive when the capture triggers.

| Option | Description |
|-----------------------------------|---|
| 0 – Standard Position | Captures the motor position in drive units. |
| 1 – Drive Internal Time | Captures the time of the trigger in ns. |
| 2 – Distributed Clock Time | Captures the network (Ethercat) distributed clock time. |
| 3 – Primary Encoder Signal | Captures the motor position triggering on primary encoder index. This mode automatically rearms after each trigger. |

If either **0 - Standard Position** or **3 - Primary Encoder Signal** is selected, delays may occur and are associated with feedback devices that are digital or interpolated .

7.8.2.0.4 Arming and Retrieving the Capture Value (CAP0.T, CAP1.T)

CAP0.EN arms the capture and CAP0.T retrieves the capture value. Once you have configured the capture, you must arm it before it will trigger. Click **Arm** (1) to arm the capture.

Once the capture is armed, when it triggers, the captured value will be displayed below the Arm button (2).

7.8.2.0.5 Setting the Capture Edge (CAP0.EDGE, CAP1.EDGE)

The capture edge determines which input state change triggers the capture.

| Option | Description |
|-------------------------|---|
| 1 – Rising Edge | Captures when the input signal goes high, from a low state. |
| 2 – Falling Edge | Captures when the input signal goes low, from a high state. |
| 3 – Both Edges | Captures any time the input signal changes state. |

7.8.2.0.6 Setting the Pre-Condition Event: (CAP0.EVENT, CAP1.EVENT)

The Capture Pre-Condition Event gives the user more flexibility in setting what conditions must be present for the Capture to trigger.

| Event Option | Description |
|--|--|
| 0 – No precondition | Capture triggers as soon as the capture edge occurs. |
| 1 – Trigger Edge after precondition | Captures triggers only when the precondition occurs before the capture edge occurs. |
| 2 – Trigger Edge while precondition = 1 | Captures triggers only while the precondition is evaluated and is true while the capture edge occurs. |
| 3 – Trigger Edge while precondition = 0 | Captures triggers only while the precondition is evaluated and is false while the capture edge occurs. |

7.8.2.0.7 Setting up a Pre-Condition for complex capture

Setting the Precondition Edge: (CAP0.PREEDGE, CAP1.PREEDGE)

The pre-edge determines what input state change triggers the precondition. This feature operates the same as the capture edge described above.

Setting the Pre-Condition Select: (CAP0.PRESELECT, CAP1.PRESELECT)

The preselect chooses what input source will trigger the precondition (based on the preedge setting, and the prefilter setting). This feature operates the same as the capture source described above.

7.8.3 Kollmorgen Test Reports

Position Capture test report based on performance testing by Kollmorgen:

Capture Accuracy with External Sensor

Drive: AKD PDMM-T00306-NBAN-000

Motor: AKM-21C

Feedback Type: Incremental type 2048 line encoder

Digital Input used: DIN1 (high speed input)

DIN1.FILTER = 0 (very important to set this to zero so filtering does not delay the system response)

Sensor Used : IDC RP1 type mounted directly to the shaft. Common industrial limit switch.

MOVE.RUNSPEED = 1000 RPM

With the motor running at above speed and the capture mechanism armed, the drive was able to capture the position within 30-70 counts (.17 - .40 degree) of accuracy or 27.5 - 64.0 micro sec.

Capture Accuracy with Internal Index from encoder

Drive - AKD PDMM-T00606-NBAN-000

Motor-AKM22G

Feedback Type - Incremental Encoder 2048 lines

MOVE.RUNSPEED = 1000 RPM

With the motor running at above speed and the capture mechanism armed, the drive was able to capture the position within 10-20 counts (.05 - .11 degree) of accuracy or 9.5 - 18.0 micro sec.

Related Parameters

CAP Parameters (pg 1)

8 Using AKD PDMM in a Vertical Axis

NOTE

For firmware versions 1-06 and later.

The drive should be configured to execute a controlled stop action in the event of a disable command (DRV.DISMODE = 2 or 3). This allows the motor to stop and the brake to be applied before the drive stops holding up the axis and removes power. With a vertical axis application where the motor is equipped with a brake that is controlled by AKD PDMM, it is recommended to set up the axis to immediately apply the brake in the event of the power stage disabling for any reason. By setting MOTOR.BRAKEIMM (→ p. 595) = 1 (default is 0=off) the brake will apply immediately when the power stage is disabled.

The power stage can disable as a result of:

- Hardware Enable removed
- Software Enable removed
- Safe Torque Off (STO) activated
- Fault Condition
- Controlled Stop Input – any input configured as (DINx.MODE = 13 controlled stop)

If the vertical axis is stationary and disabled, toggling the brake with MOTOR.BRAKERLS (→ p. 596) will not reapply the brake until controlled stop conditions are met (the axis actual velocity has been less than CS.VTHRESH (→ p. 332) for CS.TO (→ p. 331) milliseconds). In order for the brake to reapply immediately when commanding MOTOR.BRAKERLS, set CS.VTHRESH to a value greater than the velocity possible by the free falling axis and set CS.TO to 1 millisecond (its minimum value) prior to making the command.

NOTE

For a digital input initiated controlled stop (A fault that results in controlled stop. See Fault and Warning Messages (→ p. 177)), or a software disable that is configured for a controlled stop (see DRV.DISMODE (→ p. 385)), the brake will be applied at the end of the controlled stop when the power stage disables.

It is still important to setup the brake delays. This can be done in the brake screen in [EWV¹](#). These brake delays are used to delay the disabling of the power stage when motion is stopped and the drive is disabled. This is only pertinent for Controlled Stop and Software Disable of the drive. This prevents a slight drop of the load between the disable and the brake engagement. In other cases, (Fault, Hardware Enable, STO) hardware disconnection prevents the AKD PDMM from delaying the disable.

There is an advanced setting for the Hardware Disable. A programmable delay is available on the HW Enable input so the drive can hold the load before the disable. With a vertical load some movement could occur when the drive is disabled with a hardware input because the power stage was disabled immediately, before the brake could apply. DRV.HWENDELAY (→ p. 412) is available which will delay the disabling of the power stage. During this delay time, the drive will apply the brake immediately and simultaneously attempt to follow normal disable procedures. The value is limited to a maximum of 167 ms. During this time, the drive will attempt to bring the load velocity to 0 RPM. If DRV.HWENDELAY = 0, the feature is off (this is default).

¹Embedded Workbench Views

| Vertical Application Desired Response | Customer Disable Condition | Configuration Needed | Parameter Settings Needed | |
|--|-----------------------------|--|--|--------------------|
| Controlled stop when possible. If not possible, apply motor brake immediately. | Normal disable | Command CS using input. Do not disable power stage or command other disables until CS completes. | DRV.DISMODE = 2. Controlled stop then disable. Set: DRV.DISTO , CS.VTHRESH , CS.DEC , CS.TO . | |
| | E-stop, or safety condition | | | |
| | Fault CS | Do not disable power stage or command other disables until CS completes. | | |
| | Fault Dynamic Brake | N/A | | N/A |
| | Fault Power stage disable | None | | MOTOR.BRAKEIMM = 1 |
| Controlled stop when possible. If not possible, dynamic brake. | Normal disable | Command CS using input. Do not disable power stage or command other disables until CS completes. | DRV.DISMODE = 3. Controlled stop, then dynamic brake. Set: DRV.DISTO , CS.VTHRESH , CS.DEC , CS.TO . | |
| | E-stop, or safety condition | | | |
| | Fault CS | Do not disable power stage or command other disables until CS completes. | | |
| | Fault Dynamic Brake | N/A | | N/A |
| | Fault Power stage disable | None | | MOTOR.BRAKEIMM = 1 |

9 Configuring with Linear Motors

9.1 Connecting a DDL Motor to an AKD PDMM Drive

Before Connecting a DDL motor to an AKD PDMM drive, the following tasks must be accomplished:

- Integrate motor coil and magnet way onto a bearing structure so that the motor moves freely (rubber stops at the end of travel are recommended, especially during commissioning).
- Linear scale is integrated to the assembly and set up with the correct alignment and airgap to provide an appropriate sinusoidal or digital feedback signal.
- Determine the resolution of the Linear Scale in micrometers (microns) per cycle (this will be listed in the documentation of the linear scale).

Connect Hall sensor, Linear Scale, and motor temperature cables through the ACI-AKD PDMM cable assembly to the AKD PDMM's X10 Feedback Connector.

Connect the motor power leads to the AKD PDMM motor power connector X2 with the following connections:

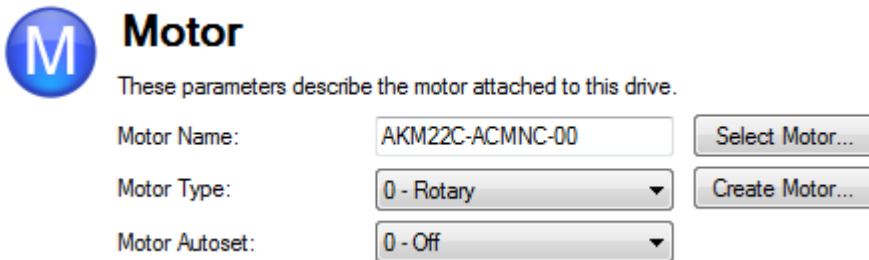
Red -> U

White -> V

Black -> W

Yellow / Green -> PE

Apply 24 volt logic power to the AKD PDMM and launch KAS IDE from a computer to interface with the AKD PDMM drive. From the Motor view in [EWV¹](#), click Select Motor.



M **Motor**

These parameters describe the motor attached to this drive.

Motor Name:

Motor Type:

Motor Autoset:

NOTE

If "Select Motor" is grayed out, Motor Autoset may need to be set to "0 -Off" to enable the Select Motor option.

On the Select Motor screen, for Motor Family select either **IC and ICD Series Ironcore DDL** or **IL Series Ironless DDL**. On the Select Motor screen for "Name" select the appropriate motor part number.

¹Embedded Workbench Views

Select Motor

M Motor
Select the motor that is attached to the drive.

To attach a motor, first select Motor family and then Motor series.

Motor Family:

AKM Series
 AKM Series
 VLM Series
 C and CH Series Cartridge DDR
 D and DH Series DDR
 KBM Series
IC and ICD Series Ironcore DDL
 IL Series Ironless DDL
 Custom Motors...

Connectors:
 M

Click OK.

From the main tree, under Settings, select **Feedback 1**.

Under Feedback Selection, select either **10 – Incremental Encoder with Halls**, or **20 – Sine Encoder with Halls** to correspond to the Linear Scale that is integrated with the motor.

Feedback 1 (X10)
The primary position feedback fitted to your motor.

Feedback Selection: -1 - Auto Identify

Feedback Identified: -1 - Auto Identify
10 - Incremental Encoder with Halls
11 - Incremental Encoder without Halls
20 - Sine Encoder with Halls
21 - Sine Encoder without Halls
30 - Endat 2.1
31 - Endat 2.2
32 - BiSS Analog
33 - Hiperface
34 - BiSS Mode C Renishaw
40 - Resolver
41 - SFD (Smart Feedback Device)
42 - Tamagawa Serial
43 - Network Cyclic Feedback
44 - Network Cyclic Feedback via FB2
50 - Reserved (do not use)
63 - Simulated Motor

If you rotate the motor shaft you should see the dial move.

Motor Autoset: 0

Position Feedback: 0 Counts 16Bit

Drive Direction: 0

Position Feedback Poles: 2

[Goto Wake And Shake Configuration](#)

Using the resolution of the Linear Scale in Microns per cycle, the Sine Cycles/Magnet Pitch is determined. Use the following:

1. Take the reciprocal of resolution to get cycles per micron
2. Multiply by 1000 to get cycles per millimeter
3. Multiply by 32 millimeters per Magnet Pitch to get Sine Cycles/Magnet Pitch

For example, if the resolution of the Linear Scale is 40 microns per cycle, then the Sine Cycles/Magnet Pitch would be 800.

The Linear Scale phase direction must be verified. Watch the Feedback 1 screen in EKV. When the coil assembly is moved in the direction of the cable exit (think of pulling the coil by the cable), the Position Feedback should increase positively in value and in the motor graphic, the gray block should move to the right. If the direction is opposite, then the A+ and A- signals on the Linear Scale must be swapped to correct the phase direction.

The motor is now ready for velocity loop and position loop compensation.

10 Configuring General Drive Settings

| | |
|--|-----------|
| 10.1 Limits | 74 |
| 10.2 Compare Engines | 75 |
| 10.3 Enable/Disable | 77 |
| 10.4 Controlled Stop | 80 |
| 10.5 Dynamic Braking | 82 |
| 10.6 Emergency Stop | 82 |
| 10.7 Under Voltage Fault Behavior | 84 |

10.1 Limits

This screen allows you view and modify the various drive limits.

| Button or Dialog Box | Description | Parameter |
|----------------------------|---|--------------------------|
| Current Limits | | |
| Positive Peak Current | The maximum positive current allowed. | IL.LIMITP (→ p. 575) |
| Negative Peak Current | The maximum negative current allowed. | IL.LIMITN (→ p. 574) |
| Dynamic Break Peak Current | The maximum amplitude of the current for dynamic braking. | DRV.DBILIMIT (→ p. 378) |
| Velocity Limits | | |
| Positive Speed Limit | The maximum speed allowed in the positive direction. | VL.LIMITP (→ p. 748) |
| Negative Speed Limit | The maximum speed allowed in the negative direction. | VL.LIMITN (→ p. 746) |
| User Over Speed Limit | The user set threshold velocity for an over speed fault. | VL.THRESH (→ p. 753) |
| Overall Over Speed Limit | The overall over speed fault threshold. | VL.VFTHRESH (→ p. 755) |
| Position Limits | | |
| Maximum Position Error | The maximum position error. If the position error PL.ERR is larger than PL.ERRFTHRESH, then the drive generates a fault | PL.ERRFTHRESH (→ p. 639) |
| Position Limit 0 | The minimum position the drive can reach before generating a negative software position fault. | SWLS.LIMIT0 (→ p. 702) |
| Position Limit 1 | The maximum position the drive can reach before generating a positive software position fault. | SWLS.LIMIT1 (→ p. 703) |
| Acceleration Limits | | |
| Acceleration | The acceleration ramp used to profile some types of motion. | DRV.ACC (→ p. 366) |
| Deceleration | The deceleration ramp used to profile some types of motion. | DRV.DEC (→ p. 379) |
| Motor Limits | Motor limits are set through the Motor Foldback Screen (see Foldback (→ p. 59)) | |

10.1.1 Limits

The limits screen covers most of the basic system limits, including current, velocity, and position.

- **Current Limits:** The current limits are set based on the drive ratings. You can change these limits to be lower than the default values for the drive, however, this may effect the expected performance of your application.
- **Velocity Limits:** The velocity limits are set based on the motor ratings. You can modify these settings above the ratings of the motor if the application requires some overhead, but be aware the motor has

mechanical limitations and may be damaged if run above those limits. It is best to leave these at the default ratings for the motor selected.

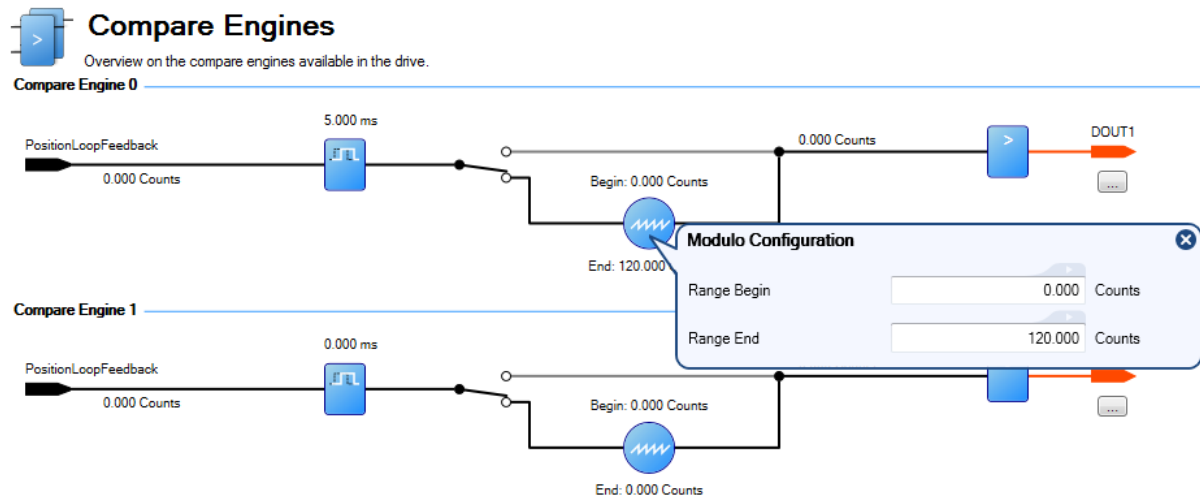
- **Position Limits:** The position limits can be set based on your specific machine application requirements. The Maximum Position Error can be set to cause a fault when the position error exceeds the value you insert here. Each limit switch can be either the lower or the upper software limit switch register, depending on the value of SWLS.EN. If only one limit switch is enabled (SWLS.EN = 1 or 2), then SWLS.LIMIT0 is the lower limit, and SWLS.LIMIT1 is the upper limit. If both limit switches are enabled (SWLS.EN = 3), then whichever switch is set to a larger value is the upper limit switch. The other switch becomes the lower limit switch. As the motor reaches the lower limit, the motor will stop and display the warning n107. As the motor reaches the upper limit, the motor will stop and display the warning n108.
- **Acceleration Limits:** This field allows you to raise accelerations to give the system crisp moves. These limits default to a low value, so you may wish to change these after the mechanics and other sections of your system are defined.

10.2 Compare Engines

The AKD PDMM Position Compare feature allows the drive's digital outputs to be turned on or off depending on one or more feedback positions. The AKD PDMM supports two independent compare engines (CMP0, CMP1). The comparison is done at a hardware level, making it faster than Programmable Limit Switch (→ p. 1), which is done through firmware.

10.2.1 Configuring Position Compare

Select Compare Engines under Settings in the [EWV¹](#) navigation tree. This view shows an overview of the two compare engines. Use this view to set each engine's compare source, output, phase advance, and modulo settings.



10.2.1.1 Order of Configuration

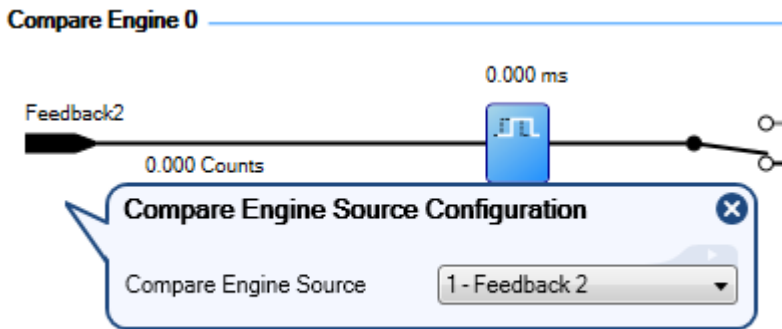
The compare parameters, CMPx.SOURCE, CMPx.MODBOUND1, CMPx.MODBOUND2, CMPx.WIDTH and CMPx.MODEN must be set prior to arming setpoints with CMPx.ARM. If CMPx.SETPOINT is not set prior to CMPx.MODEN or CMPx.ARM an error message will not be generated, but the compare will trigger at either the last set value or the default value of 0 if the drive has been power cycled or rebooted.

Prior to enabling modulo functionality with CMPx.MODEN, CMPx.SOURCE, CMPx.MODBOUND1, CMPx.MODBOUND2, and CMPx.WIDTH must all be set.

10.2.1.2 Setting Compare Sources (CMPx.SOURCE (→ p. 311))

¹Embedded Workbench Views

Compare setpoints are compared against compare sources. If they match, an output pulse is generated. Use CMP0.SOURCE and CMP1.SOURCE to select the source which the compare setpoints will be compared against.



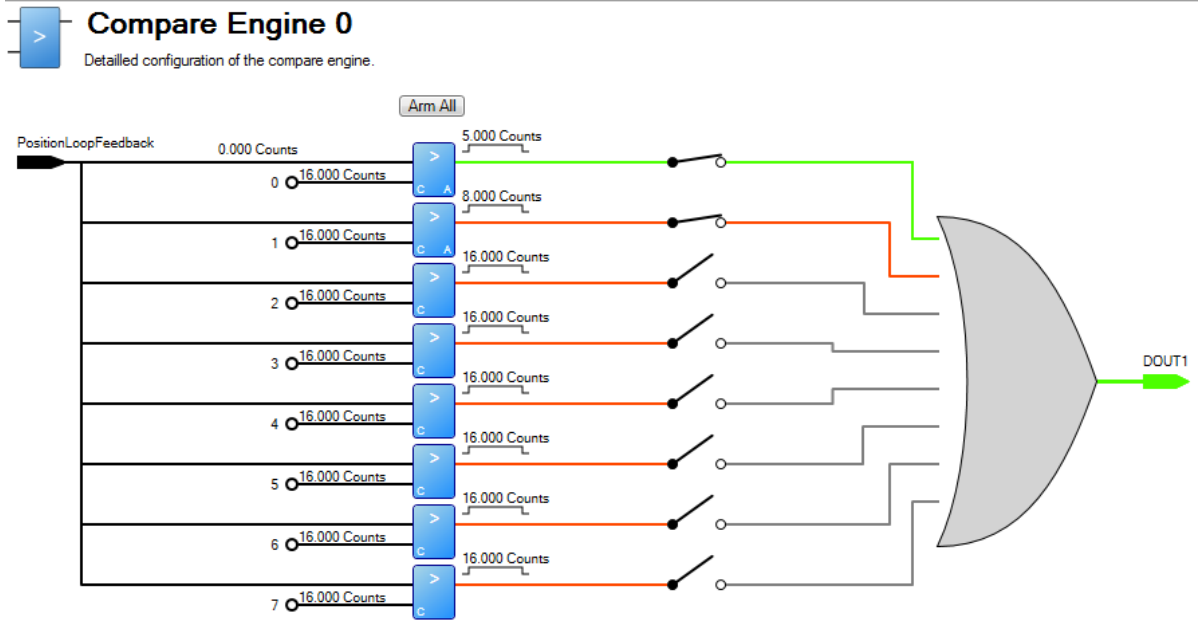
Compare Engine 1

Select one of the following compare source values:

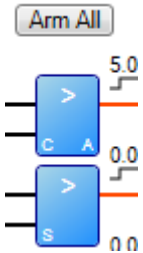
| Value | Description |
|-------|---|
| 0 | Triggers output based on a compare with the primary feedback position (FB1.P (→ p. 480)), independent of the value of PL.FBSOURCE (→ p. 645). The units of the primary feedback are determined by FB1.PUNIT (→ p. 489). |
| 1 | Triggers output based on a compare with the secondary feedback position (FB2.P (→ p. 1)), independent of the value of PL.FBSOURCE. The units of the secondary feedback are determined by FB2.PUNIT (→ p. 1). |
| 2 | Triggers output based on a compare with the tertiary feedback position (FB3.P (→ p. 507)), independent of the value of PL.FBSOURCE. The units of the tertiary feedback are determined by FB3.PUNIT (→ p. 512). |
| 3 | Triggers output based on the position of PL.FB (→ p. 644). The units of the position feedback are determined by UNIT.PROTARY (→ p. 1) or UNIT.PLINEAR (→ p. 1). |

10.2.2 Setting Compare Setpoints (CMPx.SETPOINT (→ p. 309))

Each compare engine has eight independent setpoints that can trigger output pulses with widths in either position or time. Use a compare engine's view to set each setpoint's value, width, width-type, mode, and to arm each setpoint.



Each setpoint has either a C or S in its lower left corner, indicating either continuous or single-shot mode.



Setpoints can be armed using `CMPx.ARM`. In EWV, all setpoints can be armed for a given engine using the **Arm All** button. If a setpoint is armed, an A will appear in the bottom right corner of its graphic.

Related Topics

CMP Parameters (→ p. 297)

10.3 Enable/Disable

10.3.1 Enable Modes

The AKD PDMM offers several options for hardware and software enables, as well as safe torque off (STO) to cover a variety of conditions.

10.3.1.0.1 Hardware Enable Mode

The AKD PDMM has two hardware enable methods. These methods are controlled by `DRV.HWENMODE`. Mode 0 allows for the drive to enable and clear faults on the rising edge of the hardware enable input. Mode 1 will NOT clear any faults on the rising edge of the hardware enable input, allowing the review of any current faults for manual clearing.

10.3.1.0.2 Software Enable Default

In addition, the Software Enable has two methods for enabling the AKD PDMM. These are controlled by `DRV.ENDEFAULT`. Default 0 leaves the software in the disabled state upon start-up. Default 1 enables the software upon start-up.

10.3.2 Disable Modes

Use DRV.DISMODE (→ p. 385) to select the method for stopping the drive when DRV.DIS (→ p. 384) (software disable) is commanded or upon a fieldbus command.

Disable

Disable Mode:

Disable Timeout: ms

1 - Dynamic brake immediately

2 - Controlled stop then disable

3 - Controlled stop then dynamic brake

Mode 0: Immediately disable drive.

With this condition, the drive will immediately disable the power stages and the driven motor will either coast to a stop or in the case of a vertical or overhung load axis, will fall abruptly. If a brake is present, the brake will be applied according to MOTOR.BRAKEIMM (→ p. 595) and MOTOR.TBRAKEAPP (→ p. 622). Using Digital Input mode 13, you can achieve a controlled stop as described in Mode 2.

Mode 1: Dynamic Brake to a stop.

In this situation, the drive will use the dynamic brake feature and stop motion quickly, and then disable the power stage. In most cases, the driven motor will stop quickly (pending the joules available and load circumstances). In the case of an overhung or vertical load, the driven motor will attempt to stop, but then will continue to allow the load to fall if no measures have been taken to secure the load.

Mode 2: Controlled stop, then disable.

In this mode, a controlled stop will take place based on a variety of parameters that you set. First, the driven motor will decelerate at a controlled rate (CS.DEC (→ p. 329)) until one of two things happens. 1) The motor reaches the velocity limit set (CS.VTHRESH (→ p. 332)) for a period of time (CS.TO (→ p. 331)), or 2) The drive emergency time out is reached (DRV.DISTO (→ p. 388)). Once either of these cases is reached, the power stage will be disabled (and brake applied if present)

Mode 3: Controlled stop, then dynamic brake.

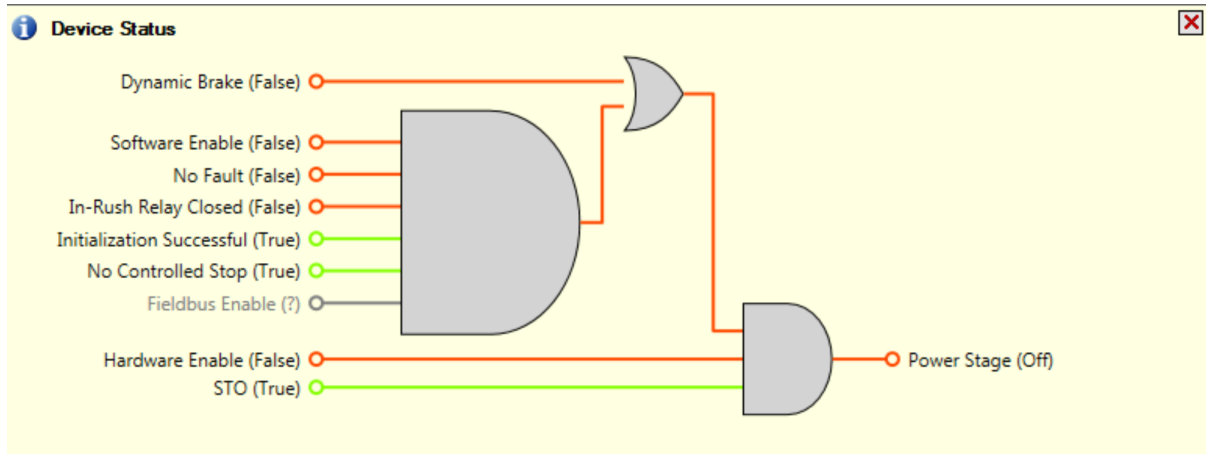
Similar to Mode 2, the motor will decelerate at a controlled rate (CS.DEC) until CS.VTHRESH is reached for a period of time (CS.TO). The drive will then dynamically brake and disable under the same conditions as described in Mode 1.

The Disable Timeout Setting determines the amount of time the drive will follow the disable mode before it deactivates the drive regardless of method chosen and alerts with an Emergency Timeout Fault.

All disable modes operate based on the type of disable command received. Any critical faults, hardware disable, or STO disable immediately turn off the power stage, and the motor will coast or free fall, depending on how the motor is secured.

10.3.3 Drive Status

The **Drive Status** area is displayed below the settings area and includes a graphical representation of drive active status with different sets of inputs. If the input or output is enabled, then it is shown in green; if the input or output is disabled, then it is shown in red. The inputs to the OR and & (AND) gates identify which conditions are true (green) or false (red) and follow the normal logic for & (AND) and OR gates. This diagram is useful for finding which input may be preventing the drive from being enabled.



10.3.4 Controlled Stop

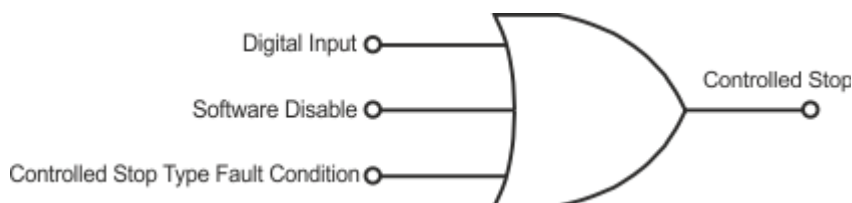
The Controlled Stop area displays values for parameters associated with controlled stop.

| Button or Dialog Box | Description | Parameter |
|----------------------------|---|-----------------------|
| Velocity Threshold | Sets the velocity threshold for controlled stop process. | CS.VTHRESH (→ p. 332) |
| Velocity Threshold Timeout | Sets the velocity threshold, which is the time value for the drive velocity to be within CS.VTHRESH before the drive disables | CS.TO (→ p. 331) |
| Deceleration | Sets the deceleration value for the controlled stop process. | CS.DEC (→ p. 329) |
| Control stop Input | Displays the list of digital inputs configured to controlled stop mode separated by ','. When no controlled stop mode is configured, this box displays the message: No CS Input Configured . The Configure Input link opens the Digital I/O screen where you can configure the controlled stop mode. | |

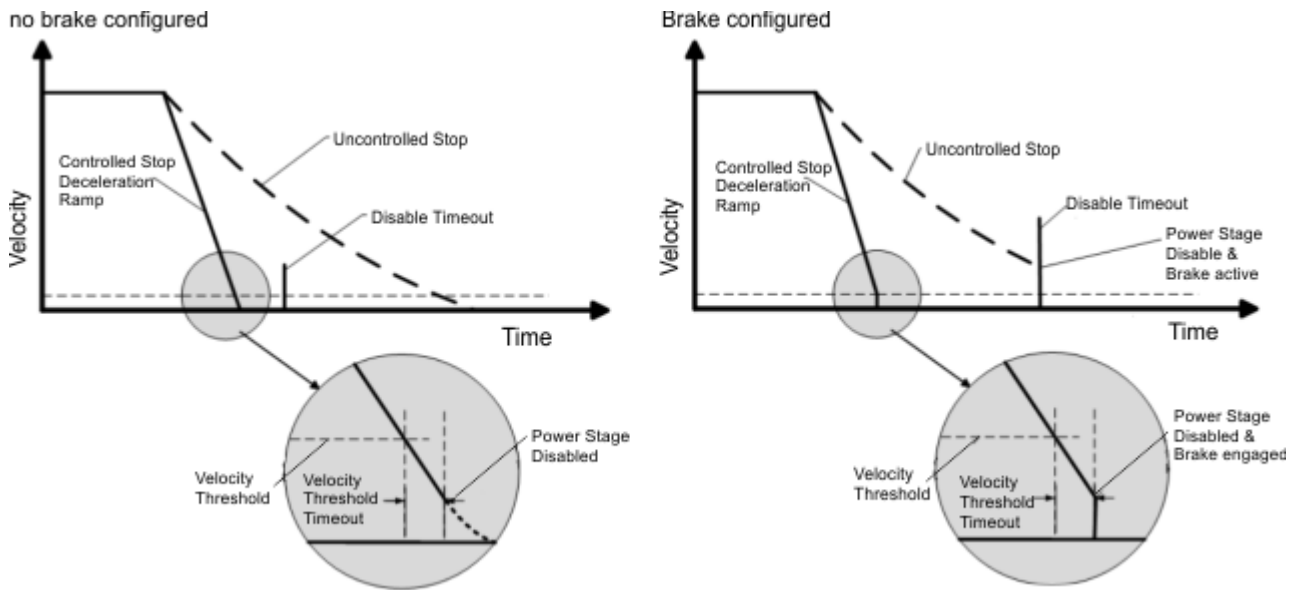
10.3.5 More/Less Button

The **More** button reveals the status diagram for the controlled stop configuration. It also displays the block diagram for the control stop. Two block diagrams are available: one for a brake fitted and another for no brake.

Control stop status diagram



Controlled Stop Block Diagram



10.4 Controlled Stop

In a controlled stop, drive motion is brought to a standstill in a controlled manner. The drive commands a zero velocity from the motor. The motor decelerates at the prescribed deceleration value (CS.DEC (→ p. 329)).

A controlled stop can occur in three ways:

- The user configures a programmable digital input to mode 13 using DINx.MODE. For example, if [DIN1.MODE 13](#) is applied, digital input 1 is set to controlled stop.
- Either a controller or the user (through the [EWV](#)¹ terminal window) initiates a software disable (DRV.DIS) command.
- A fault initiates a controlled stop from the drive. See [Fault and Warning Messages](#) (→ p. 177) for the faults which initiate a controlled stop.

The controlled stop mechanism is activated in the following cases:

- DRV.DISM0DE = 2 and user executes DRV.DIS from the terminal or EWV disable buttons.

NOTE

You must disable the drive in order to set DRV.DISM0DE.

- DRV.DISM0DE = 2 and user executes DRV.DIS from a fieldbus connected to the drive.
- A fault occurs which has a controlled stop (CS) reaction. After the CS is executed, the drive disables.
- A digital input mode (DINx.MODE) is set to 13. If the digital input state changes (active high or low according to DINx.INV) the CS is executed, and then the drive disables.
- HW limit switch: A digital input is defined as a positive (negative) limit switch (DINx.MODE 18 or 19). When the limit switch is met, the CS mechanism starts running. In this case, the parameter DRV.DIST0 is not active.
- SW limit switch: SWLS defines an active SW limit. When the limit is met, the CS mechanism starts running. In this case, the parameter DRV.DIST0 is not active.

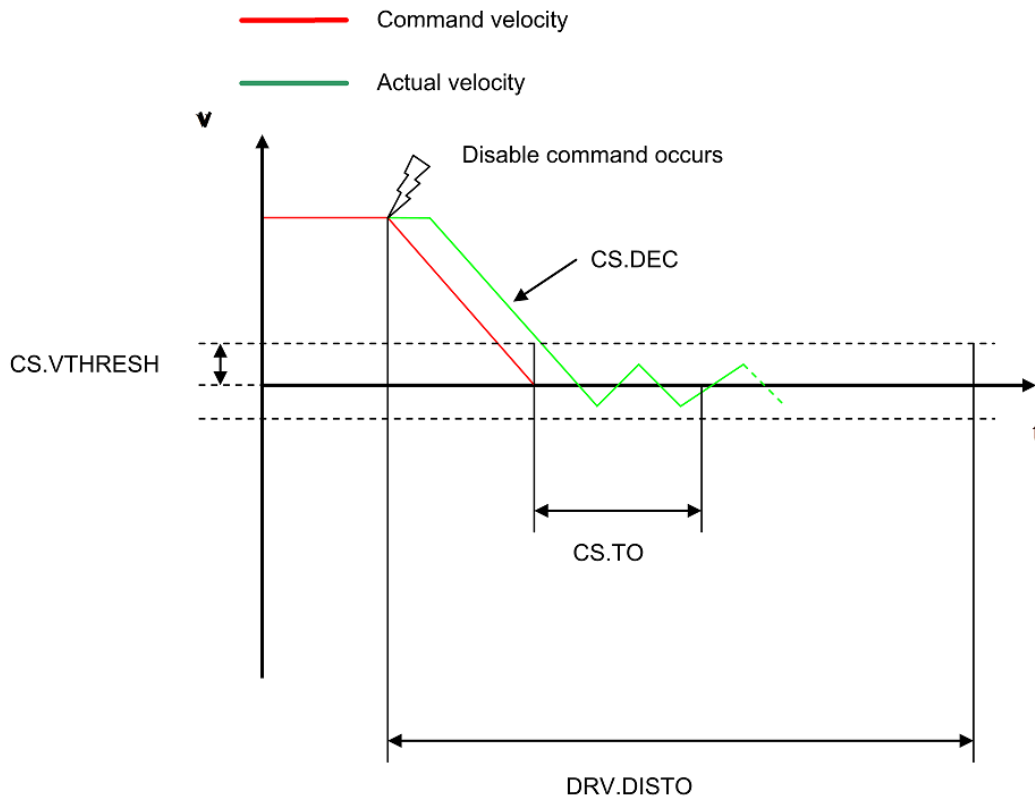
Use the drive CS parameters to configure a controlled stop as follows:

1. CS.DEC: Deceleration ramp that is used for disable.
2. CS.VTHRESH: Velocity 0 threshold. The motor shaft is considered as stopped as soon as the actual velocity (filtered through a 10 Hz filter, such as VL.FBFILTER) is within \pm CS.VTHRESH.

¹Embedded Workbench Views

3. CS.TO: Velocity 0 time. The actual velocity must be consecutively within $0 \pm \text{CS.VTHRESH}$ for the time CS.TO, before the drive completes the CS process. This value is used since the motor can overshoot out of the VELD window depending on the gains, deceleration ramp, motor inertia and so on.
4. DRV.DISTO: Disable time out. This parameter sets an overall and independent running check as to whether or not the drive can achieve the disable state. If the VELD window set in step 3 is too small, it is possible that the drive may never reach the end of the CS process. The DRV.DISTO parameter and functionality addresses this issue by disabling the drive after the DRV.DISTO time elapses, even if the CS process did not end.

Controlled Stop Diagram



When configuring the controlled stop feature, please note the following:

- If the HW limit switch is active and any of the other CS activated, the only difference will be that in this case the DRV.DISTO will limit the time before disabling the drive.
- If the value of DRV.OPMODE of the drive is current mode, the drive will not execute the CS but instead stop immediately.
- Set DRV.DISTO to an appropriate value that will allow the motor to decelerate from any velocity to 0 with DRV.DEC. This value must also allow the motor to afterwards remain within VL.FB for CS.TO consecutively within $0 \pm \text{CS.VTHRESH}$.

The drive issues a fault FF703 (→ p. 193) in case that the DRV.DISTO counter expires during a controlled stop procedure.

Related Parameters and Commands

CS Parameters (→ p. 328)

CS.STATE (→ p. 330): Reads the current state of controlled stop process (0 = controlled stop is not occurring. 1 = controlled stop is occurring).

DIN1.MODE TO DIN7.MODE (→ p. 340)

DRV.DIS (→ p. 384)

DRV.DISTO (→ p. 388)

DRV.DISMODE (→ p. 385)

Related topics:

Emergency Stop (→ p. 82)

1 Digital Inputs and Outputs

Fault and Warning Messages (→ p. 177)

10.5 Dynamic Braking

Dynamic braking is a method to slow a servo system by dissipating the mechanical energy in a resistor driven by the motor back EMF. The drive has a built in advanced dynamic braking mode which operates fully in hardware. When activated, the drive shorts the motor terminals in phase with the back EMF (q axis) but continues to operate the non-force producing current loop (d-axis) with 0 current. This action forces all of the dynamic braking current toward stopping the motor current and insures the fastest stopping of motor terminal current.

The drive hardware also limits the maximum dynamic braking motor terminal current via the DRV.DBILIMIT parameter to prevent the drive, motor, and customer load from encountering excessive currents/forces. When the current is not being limited, the mechanical energy is dissipated in the motor terminal resistance. When the current is being limited, energy is returned to the drive bus capacitors. When the amount of returned energy raises the bus capacitor voltage enough, the drive activates the regeneration control to start transferring the returned energy to the regen resistor. This resistor could be internal or external to the drive depending on drive model and drive wiring.

Whether and how the drive uses dynamic braking mode depends on the drive disable mode (DRV.DISMODE) setting.

10.5.1 Drive Regeneration

When the servo motor is slowing down at a rate faster than friction and motor losses would slow the motor, then mechanical energy can be returned to the drive. This returned energy initially drives the internal bus voltage upwards. When the returned energy is high enough, the regeneration control transfers the excess returned energy into the regeneration power resistor. If the regeneration control cannot fully handle the power returned (for example, because there is not a regen resistor present or its resistance value is too high), then the bus voltage will continue to rise and a bus over voltage fault will happen and disable the drive completely, which allows the motor to freewheel.

10.5.1.0.1 AKD-M00306 to AKD-M00606

These units do not have an internal regeneration resistor. In many applications machine friction, motor losses, and the limited bus capacitor energy absorption handles the application. But, depending on the exact application requirements, an external resistor can be connected.

10.5.1.0.2 AKD-M01206 to AKD-Mzzz07

These units have an internal regeneration resistor plus the capability to connect an external resistor if higher power levels are needed.

Related Topics

See section 6.14 *Dynamic Braking* in the *AKD PDMM Installation Manual* for detailed information on dynamic braking.

DRV.DISMODE (→ p. 385)

DRV.DBILIMIT (→ p. 378)

10.6 Emergency Stop

10.6.1 Stop / Emergency Stop / Emergency Off

The control functions Stop, Emergency Stop and Emergency Off are defined by IEC 60204. Notes for safety aspects of these functions can be found in ISO 13849 and IEC 62061.

The parameter DRV.DISMODE must be set to 2 to implement the different stop categories. Consult the *EWV¹ Online Help* for configuring the parameter.



WARNING

No functional safety!

Serious injury could result when the load is not properly blocked. With vertical load the load could fall.

1. Functional safety, e.g. with hanging load (vertical axes), requires an additional mechanical brake which must be safely operated, for example by a safety control.
2. Set parameter MOTOR.BRAKEIMM to 1 with vertical axes, to apply the motor holding brake immediately after faults or Hardware Disable.

10.6.1.1 Stop

The stop function shuts down the machine in normal operation. The stop function is defined by IEC 60204.

The Stop Category must be determined by a risk evaluation of the machine.

Stop function must have priority over assigned start functions. The following stop categories are defined:

Stop Category 0

Shut-down by immediate switching-off the energy supply to the drive machinery (this is an uncontrolled shut-down). With the approved safety function STO the drive can be stopped using its internal electronics (IEC 62061 SIL2).

Stop Category 1

A controlled shut-down, whereby the energy supply to the drive machinery is maintained to perform the shut-down, and the energy supply is only interrupted when the shut-down has been completed.

Stop Category 2

A controlled shut-down, whereby the energy supply to the drive machinery is maintained.

Stop Category 0 and Stop Category 1 stops must be operable independently of the operating mode, whereby a Category 0 stop must have priority.

If necessary, provision must be made for the connection of protective devices and lock-outs. If applicable, the stop function must signal its status to the control logic. A reset of the stop function must not create a hazardous situation.

¹Embedded Workbench Views

10.6.1.2 Emergency Stop

The Emergency Stop function is used for the fastest possible shutdown of the machine in a dangerous situation. The Emergency Stop function is defined by IEC 60204. Principles of emergency stop devices and functional aspects are defined in ISO 13850.

The Emergency Stop function will be triggered by the manual actions of a single person. It must be fully functional and available at all times. The user must understand instantly how to operate this mechanism (without consulting references or instructions).

The Stop Category for the Emergency Stop must be determined by a risk evaluation of the machine.

In addition to the requirements for stop, the Emergency Stop must fulfil the following requirements:

- Emergency Stop must have priority over all other functions and controls in all operating modes.
- The energy supply to any drive machinery that could cause dangerous situations must be switched off as fast as possible, without causing any further hazards (Stop Category 0) or must be controlled in such a way, that any movement that causes danger, is stopped as fast as possible (Stop Category 1).
- The reset must not initiate a restart.

10.6.1.3 Emergency Off

The Emergency Off function is used to switch-off the electrical power supply of the machine. This is done to prevent users from any risk from electrical energy (for example electrical impact). Functional aspects for Emergency Off are defined in IEC 60364-5-53.

The Emergency Off function will be triggered by the manual actions of a single person.

The result of a risk evaluation of the machine determines the necessity for an Emergency Off function.

Emergency Off is done by switching off the supply energy by electro-mechanical switching devices. This results in a category 0 stop. If this stop category is not possible in the application, then the Emergency Off function must be replaced by other measures (for example by protection against direct touching).

10.7 Under Voltage Fault Behavior

You can adjust the conditions for an under voltage fault using `VBUS.UVMODE` in the EWV¹ Terminal (→ p. 169) View:

VBUS.UVMODE = 1 (default)

The drive will not report an under voltage fault unless the drive is enabled and `VBUS.VALUE` falls below `VBUS.UVFTHRESH`

VBUS.UVMODE = 0

The drive will report an under voltage condition any time `VBUS.VALUE` falls below `VBUS.UVFTHRESH`.

When an under voltage fault occurs, the drive is disabled and issues the following alerts:

- EWW alert: 502 Bus Under Voltage
- Drive LED alert: Left LED displays [F], right LED displays [u-V].\
- Fault relay output turns on.

10.7.1 Current Loop Gain Scheduling

¹Embedded Workbench Views

Current loop gain scheduling is an advanced feature that allows the current loop gain to be adjusted for motors with varying inductance. A 256-element lookup table is used to select a gain adjustment scale factor based on current command.

This feature is needed when the motor inductance is saturated during normal operation. Because the current loop gain is calculated using the motor inductance, if the inductance changes, the current loop is at risk of becoming unstable.

If a motor is being used under high loads, the inductance is probably saturating if one or more of the following occurs:

- Audible chirping occurs.
- Chirping gets louder with higher commanded current.
- Instability occurs under high current loads (at or near MOTOR.IPEAK (→ p. 607))

To fix this problem, gain scheduling can be used to change the current loop gain as a function of the measured current, IL.FB (→ p. 555).

10.7.1.1 Using the Gain Scheduling View in EWV

To use this feature effectively, you should have either an inductance chart of motor inductance as a function of current, or you must have an idea of how the current loop is performing.

If a chart of motor inductance versus current command is available, it is possible to calculate the required current loop gain values over the range of motor currents.

Current Loop Gain = $2000 * 2\pi * \text{Motor Inductance (H)}$


Example

Motor Inductance is 3.19 mH, current loop gain would be 40.01

Current Loop Gain = $2000 * 2\pi * 0.00319 = 40.01$

Once suitable values have been identified, you can enter these values into EWV¹ in the **Gain Scheduling** view. The **Import** and **Export** buttons at the bottom of the view allow you to import and export data as .csv files.

¹Embedded Workbench Views



Gain Scheduling

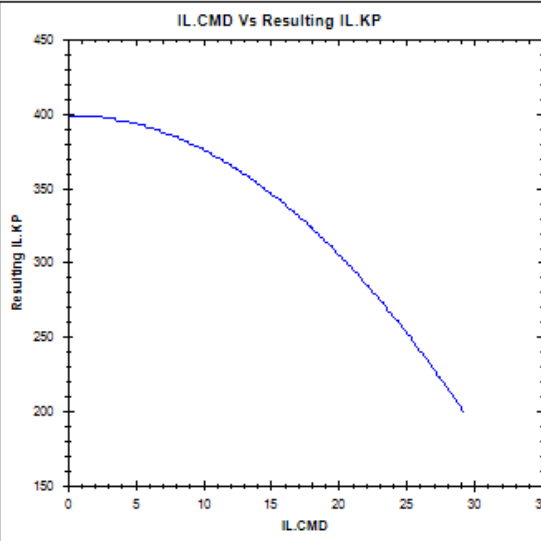
[? Learn more about this topic](#)

Advanced user current loop gain lookup table for special motors that have varying inductance.

Drive IPeak : Ams

Proportional Gain : V/A

| IL.CMD (A) | Table Index | Value (%) | Resulting IL.KP |
|------------|-------------|-----------|-----------------|
| 0.000 | 0 | 100.000 | 400.072 |
| 0.115 | 1 | 99.607 | 398.500 |
| 0.229 | 2 | 99.607 | 398.500 |
| 0.344 | 3 | 99.607 | 398.500 |
| 0.459 | 4 | 99.607 | 398.500 |
| 0.573 | 5 | 99.607 | 398.500 |
| 0.688 | 6 | 99.607 | 398.500 |
| 0.803 | 7 | 99.607 | 398.500 |
| 0.917 | 8 | 99.607 | 398.500 |
| 1.032 | 9 | 99.607 | 398.500 |
| 1.146 | 10 | 99.607 | 398.500 |
| 1.261 | 11 | 99.607 | 398.500 |
| 1.376 | 12 | 99.607 | 398.500 |



By default, the current loop value will be what is defined in IL.KP across the whole range of currents. To change the value of IL.KP over a range of values, simply enter a scaling term: 0 - 100% of the current value.

Example

If the current loop gain entered on IL.KP was 40.124 (shown above) and a current loop gain of 36 was desired, a scaling term of 90% should be entered for the desired current ranges.

$$40.124 * 0.90 = 36.112$$

10.7.1.1.1 Using the Terminal View for Gains Scheduling

You can also use the Terminal to set up the Gains Scheduling Table. If you use the Terminal, then two parameters are required for each lookup table point: IL.KPLOOKUPINDEX (→ p. 569) and IL.KPLOOKUPVALUE (→ p. 570). IL.KPLOOKUPINDEX specifies the lookup table index (0 – 255), and IL.KPLOOKUPVALUE specifies the scaling term (0 – 100%) to scale IL.KP.

The current to which a lookup index refers can be calculated as follows:

$$IL.CMD \text{ Range} = DRV.IPEAK/157 * IL.KPLOOKUPINDEX$$

A full list of table values can also be retrieved using IL.KPLOOKUPVALUES (→ p. 571), which returns a comma delimited table as follows:

```
-->IL.KPLOOKUPVALUES
Index Value
0, 100.000
1, 100.000
2, 100.000
3, 100.000
4, 100.000
```

5, 100.000
6, 100.000
7, 100.000
8, 100.000
9, 100.000
10, 100.000

10.7.1.2 Rules for Populating Gain Scheduling Table

The gain scheduling table is populated according to the following rules:


1. When the drive powers up:
 - a. If MOTOR.AUTOSET=1 and FB1.MEMVER \geq 0.03, the table is read-only. The entries are calculated and set based on the value of MOTOR.LISAT (→ p. 611).
 - b. Otherwise, if table entries have been stored as non-volatile parameters, the values are restored.
 - c. Otherwise, the table entries are set to default values.
2. When DRV.RSTVAR (→ p. 433) is commanded:
 - a. Rule 1a is applied.
 - b. Otherwise, rule 1c is applied.
3. When MOTOR.AUTOSET is set to 1, rules 1a to 1c are applied in order.
4. When the lookup table is imported from a parameter file:
 - a. Rule 1a is applied (a read-only table cannot be updated).
 - b. Otherwise, table entries are set to the imported values.

11 Creating Motion

| | |
|--|-----------|
| 11.1 Service Motion | 89 |
| 11.2 Motion Profile Table | 90 |

11.1 Service Motion

Service Motion allows you to set up simple motion (Command Source must be set to Mode 0 - Service). It is typically used during initial set up to exercise the system. It can be used to help troubleshoot the system, execute tuning, make repetitive moves for verification of mechanical set up, or for other general simple motion needs. There are multiple ways to set up motion depending on the desired result in torque, velocity or position modes. In all modes, you can run a momentary pulse, set up reversing motion, or initiate continuous motion.

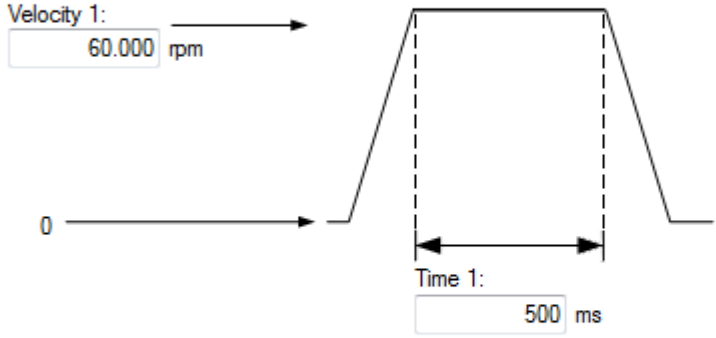
 **Service Motion**
Service motion allows you to start and stop some test motions.

Service Motion Mode: Pulse Reversing Continuous

Group:

Velocity 1: rpm

0



Time 1: ms

[Goto Limits](#)

Acceleration: rpm/s ⓘ Accel. Limit: rpm/s

Deceleration: rpm/s ⚠ Decel. Limit: rpm/s

⚠ Drive is inactive.

Position Feedback: Counts16Bit

Velocity Feedback: rpm

The chart below identifies the commands available on the **Service Motion** view:

| Button or Dialog Box | Description | Available in OP Mode |
|----------------------|--|----------------------------|
| Pulse | Alternates between the commanded current or velocity and zero velocity. You can specify the time period between the commanded value and the return to zero. Setting a time to zero will generate a continuous command. | Torque, Position, Velocity |
| Reversing | Alternates between the two commanded values. You can specify the time the command will be held in each state. | Torque, Position, Velocity |
| Continuous | Runs the commanded current or velocity continuously. | Torque, Position, Velocity |
| Group | Selects parameter group to be used with service motion. Group 1 selects parameter group for SM.MODE 0, and group 2 selects parameter group for SM.MODE 2. See SM.MODE (→ p. 690) for more details. | Torque, Position, Velocity |

| Button or Dialog Box | Description | Available in OP Mode |
|-----------------------|--|----------------------------|
| Current 1/Current 2 | Only shows while drive is in Torque mode. Sets the two different currents. | Torque |
| Velocity 1/Velocity 2 | Only shows while drive is in Velocity or Position mode. Sets the two different velocities. | Velocity, Position |
| Time 1/Time 2 | Sets the time for which the different commands are generated. Setting a time to zero generates a continuous command. | Velocity, Position |
| Start/Stop | Starts and stops the motion. | Torque, Position, Velocity |
| Acceleration | Sets the acceleration ramp for service motion if set to greater than 0. | Velocity, Position |
| Deceleration | Sets the deceleration ramp for service motion if set to greater than 0. | Velocity, Position |
| Accel. Limit | The acceleration limit set by DRV.ACC (→ p. 366). | Velocity, Position |
| Decel. Limit | The deceleration limit set by DRV.DEC (→ p. 379). | Velocity, Position |
| Position Feedback | Displays the present position of the motor. | Torque, Position, Velocity |
| Velocity Feedback | Displays the present velocity of the motor. | Torque, Position, Velocity |
| Current Feedback | Displays the present current of the motor. Only displays when drive is in Torque mode. | Torque |

Related Parameters

SM.I1 (→ p. 688)
 SM.I2 (→ p. 689)
 SM.I2 (→ p. 689)
 SM.MOVE (→ p. 693)
 SM.T1 (→ p. 694)
 SM.T2 (→ p. 695)
 SM.V1 (→ p. 696)
 SM.V2 (→ p. 697)

11.2 Motion Profile Table

Motion Profile Tables allow S-Curve and other customer specified velocity profiles. The tables are interpolated to determine the command position each sample during the move. There are two forms of interpolation, OneToOne and Profile Table that are used to create the profile actually used for motion. OneToOne interpolation uses the table for every sample during the motion. Profile Table interpolation uses the table only for acceleration and deceleration with the insertion of a constant velocity section of the move for longer moves.

There are two default tables (Table 0 and Table 1) pre-loaded on AKD PDMM drives. Table 0 is an S-Curve profile that has sinusoidal jerk, acceleration, and velocity profiles. This provides very smooth motion and can be used with either OneToOne or Profile Table interpolation. Table 1 is an asymmetric profile that can only be

used with OneToOne interpolation. This has the highest acceleration at low speeds and is an example of a customer table that may decrease overall motion times for systems with high torque at low velocity.

You can define acceleration, velocity, position, and jerk for drive motion tasks using the **Motion Profile Table** view shown below:

⚠ CAUTION

This is a beta feature. Please note that IL.KACCFF should be set to 0 when using this feature.

Motion Profile Table [Learn more about this topic](#)

Edit the motion table that are stored in the Drive.

⚠ *Beta feature - may change. Caution: IL.KACCFF should be set to 0 using this feature !*

Tables:

| ID | Name | Size | Status |
|----|------|------|--------|
| | | | |

Name:

Position:

Velocity:

Acceleration:

Jerk:

Import Export

Add Remove Up Down

Save tables to Drive Reload tables from Drive Clear tables in Drive

Reset to Default

The **Motion Profile Table** view consists of three sections:

- Grid
- Graphical Representation
- Control Buttons

11.2.1 Grid

You can modify motion profile table data using the tables grid on the left side of the view and the Name text box on the right. All profile table data is displayed in the tables grid. This data includes:

- ID: indicates the unique ID number (0 to 7) of the profile table.
- Name: indicates the name of the profile table. By default, a profile table name is 'no_name'. To edit the profile table name select one of the profile tables from the grid, and then change the profile table name by using the **Name** textbox .
- Size: indicates the total count of the profile table data.
- Status: indicates the current status of the profile table in the grid. Before saving a table into the drive, the status displays "Modified" and after saving the drive, the status displays "Sync". If you change the profile table name, then the status displays "Name Modified".

11.2.2 Graphical Representation

On the right side of the Motion Profile Table view, you can view graphical representations of acceleration, velocity, position, and jerk of the selected profile table. Graphs are plotted based on the given data of the profile table. See the screenshot below for typical graphical representations:

Motion Profile Table [Learn more about this topic](#)

Edit the motion table that are stored in the Drive.

⚠ Beta feature - may change. Caution: IL.KACCFF should be set to 0 using this feature !

Tables:

| ID | Name | Size | Status |
|----|---------|------|----------|
| 0 | no_name | 1000 | Modified |

Name:

Position:

Velocity:

Acceleration:

Jerk:

ⓘ Table data is modified. Click on Save tables to Drive button to save the t...

11.2.3 Control Buttons

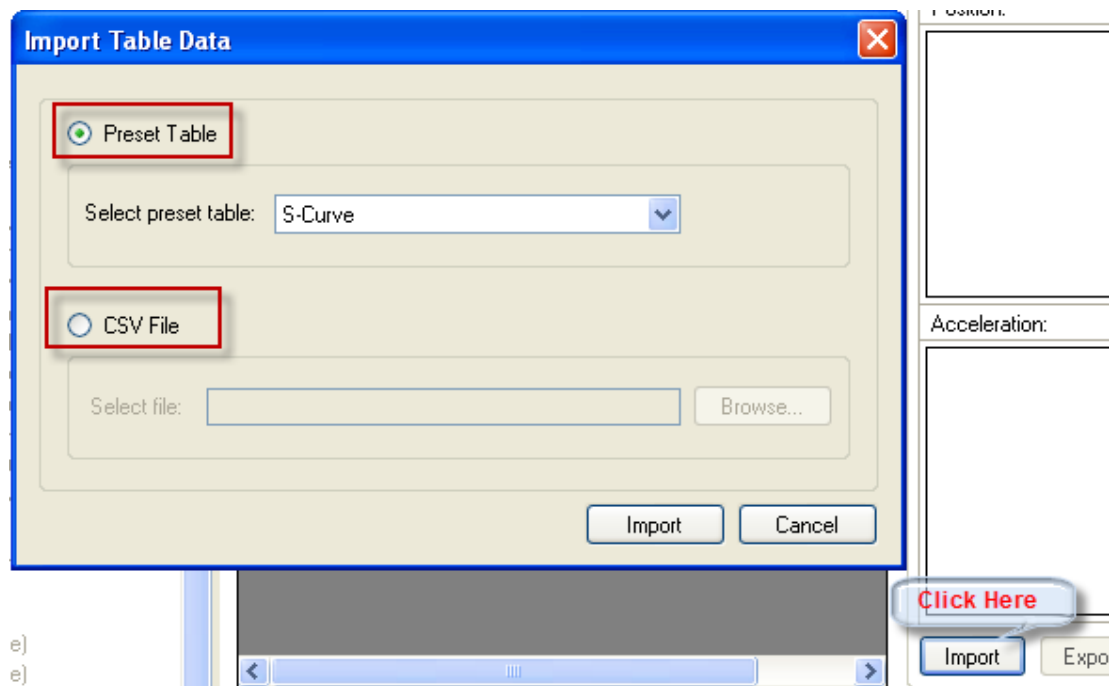
The control buttons at the bottom of the **Motion Profile Table** view allow you to manipulate motion profile tables and data as follows:

| Control Button | Description |
|--------------------------|--|
| Add | Adds new profile tables. A new table has a size of zero when it is added to the grid. You can add a maximum of 8 profile tables, numbered 0 to 7. |
| Remove | Remove profile tables from the grid (in memory). If a profile table is in use by a motion task and you attempt to delete the profile table using the Remove button, then a caution message appears. The caution message shows the affected profile table ID and an option to either continue or not. |
| CW/CCW | Moves profile table position either up or down in the grid list. If a profile table is in use by a motion task and you attempt to delete the profile table using the Up or Down buttons, a caution message appears. The caution message shows the affected profile table ID and an option to either continue or not. |
| Save tables to the Drive | Saves a newly added or modified valid profile table to the drive. The current representation in the grid is saved in the drive. |

| Control Button | Description |
|--------------------------|--|
| Reload tables from Drive | Reloads the table from the drive and overrides the table that is currently in memory. |
| Cancel | Cancel appears only when the reload operation is in progress. Click Cancel to cancel the profile table reloading operation. |
| Clear tables in Drive | Clears all the profile tables present in the drive. |
| Reset to Default Button | Loads the default table map into the memory. The current default map contains only one profile table (S-curve with 1,000 points). |
| Import | Imports the data from either preset tables within WorkBench or from an external .csv file. When you click Import , the Import Table Data popup screen appears and allows you to select the table size. |
| Export | Exports data to a .csv file. |

11.2.3.1 Import Table Data

The **Import** button opens the **Import Table Data** screen shown below:



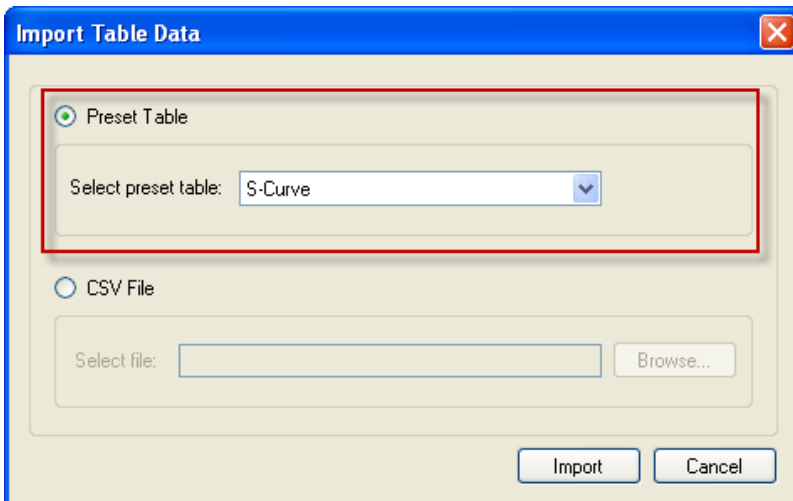
You have two options to provide inputs to the profile table:

- Preset Table
- CSV File

11.2.3.2 Importing data from Preset Table option

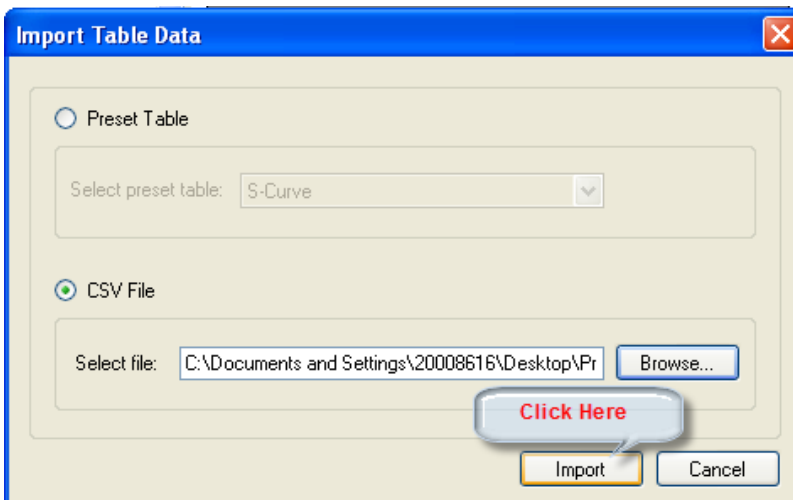
By default, the **Preset table** option is selected. You can select one of the preset tables from the **Select Preset Table** box. After selecting the source, click **Import** to complete the import of the data.

After import, the size of the profile table is shown in the grid and you can view the graphical representation of position, velocity, acceleration, and jerk based on the profile table data now loaded into the grid.



11.2.3.3 Importing data from an external CSV file

You also use the **CSV File** option to import the profile table data from an external .csv file. Select the **CSV File** option and then select the valid file using the browse button. After selecting the source file, click **Import** to complete the import of the data. A message indicates successful data import. After import, the size of the profile table is shown in the grid and you can view the graphical representation of position, velocity, acceleration, and jerk based on the profile table data now loaded into the grid.



If an invalid source or invalid format is selected, an "Invalid Data Format" message appears.

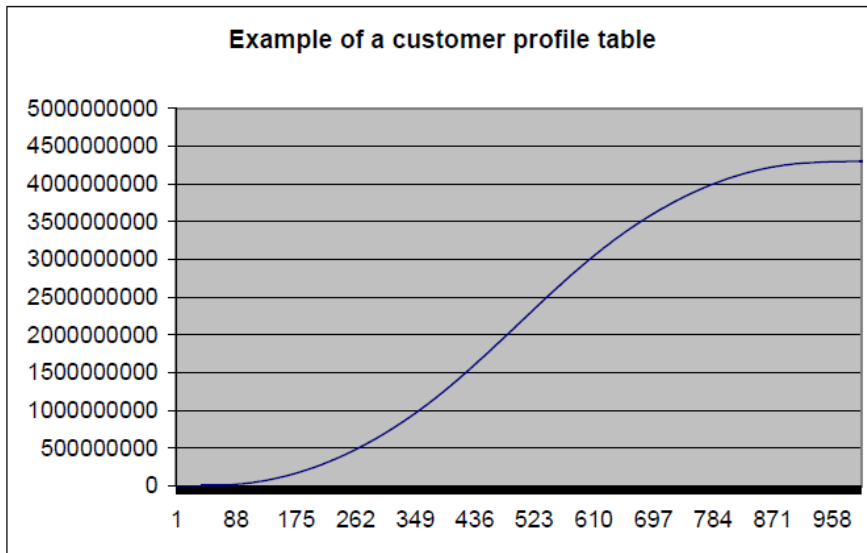
11.2.4 Creating Motion Profile Tables

A motion profile table is a table of unit-less position values, which are the integral of the velocity profile during the acceleration and deceleration process of a motion task. A motion profile can be stored in the drive and used in order to accelerate and decelerate with a certain profile shape.

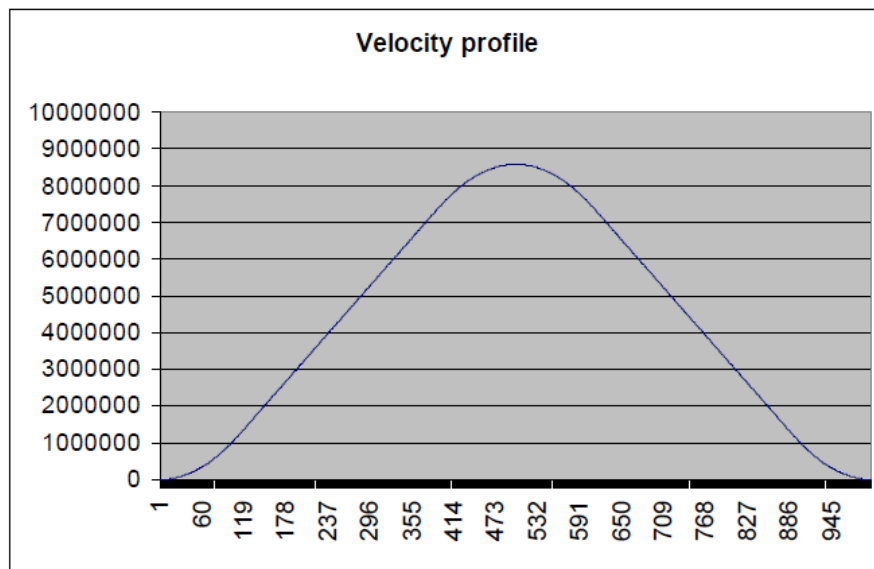
The motion profile table describes the shape of the acceleration process, but does not determine how fast the motion task accelerates or decelerates and which target velocity will be reached.

11.2.4.1 Example of a motion profile table

An example of a motion profile table is shown below:



The motion profile table is the integral of the velocity profile. The velocity profile during the acceleration and the deceleration process is shown below:



The derivative of the motion profile table is calculated using the following formula:

$$\text{velocity_profile_value}_n = \text{customer_profile_entry}_{n+1} - \text{customer_profile_entry}_n$$

11.2.4.2 Motion Profile Table Restrictions

Restrictions for motion profile tables include the following:

1. A motion profile table needs a reasonable number of entries (usually between 1,000-4,000 entries, depending on the acceleration and deceleration time of a motion task). If an acceleration or deceleration process takes more position-loop samples than half of the motion profile table entries, then the drive interpolates linearly between the single motion profile table entries.
2. The motion profile table should contain an even number of entries. The first point of the customer table starts with the value of 0 and the last point must contain the value of $2^{32}-1$.
3. The motion profile table contains values in ascending order.

- The following motion profile table entry must contain the value of nearly 2^{31} .

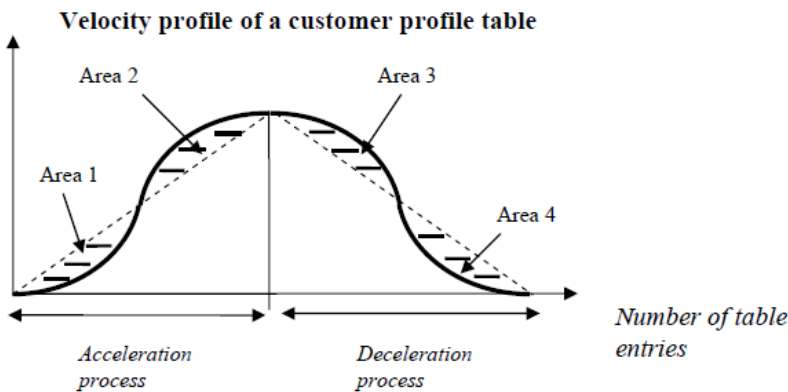
$$table_entry = \frac{number_of_table_points}{2} + 1$$

Example

Assume that a motion profile table contains 1,000 data points. In this case point $1000/2+1 = 501$ must contain the value of $2^{31} = 2,147,483,648$.

- A motion profile table must also be symmetric during the acceleration and the deceleration process when Profile Table interpolation is used.

To illustrate profile symmetry, the derivative of the motion profile table (velocity profile) is shown below. Note the symmetry according to the velocity profile.

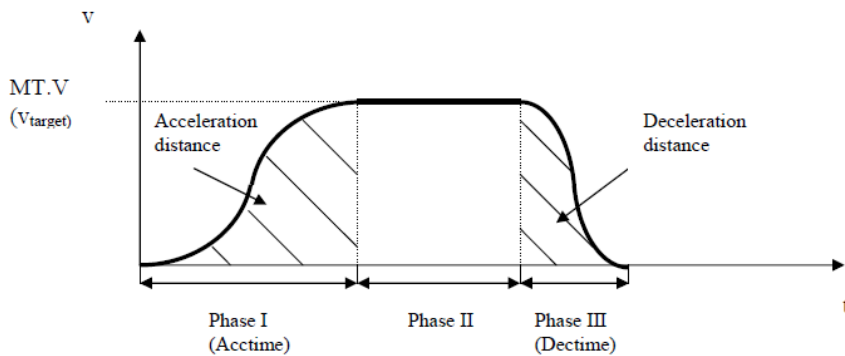


The left half of the curve describes the shape of the acceleration process of the motion task. The right half of the curve describes the shape of the deceleration process of the motion task. A symmetric motion profile table means that Area 1, Area 2, Area 3 and Area 4 have the same size.

11.2.4.3 Different methods of motion table motion tasking

11.2.4.3.1 General motion profile table explanations

The algorithm for handling the motion profile motion task are the same for both methods, the standard customer table motion task and the OneToOne customer table motion task. The diagram below illustrates a basic table profile algorithm. The figure shows a standard customer table motion task.



The drive calculates the acceleration time and deceleration time out of the given motion task parameters (see MT Parameters and Commands) with the assumption of a trapezoidal acceleration setting (MT.ACC and MT.DEC). The formulas are:

$$Acctime = \frac{MT.V}{MT.ACC} \quad ; \quad Dectime = \frac{MT.V}{MT.DEC}$$

11.2.4.4 Use of IL.KACCFF with Motion Tables

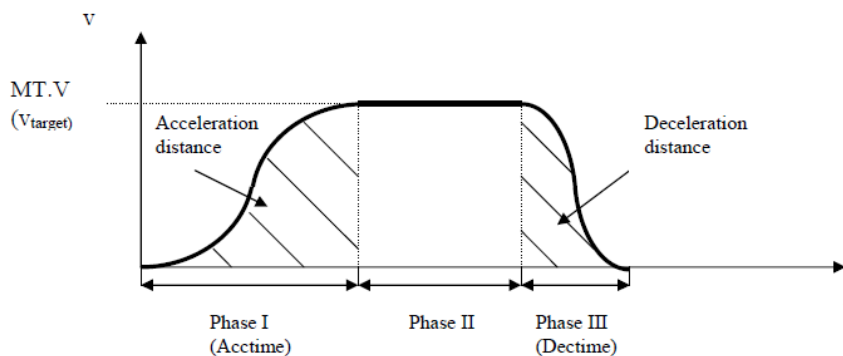
IL.KACCFF can be used with Profile Table or OneToOne interpolation motion tasks as long as the following criteria is met:

$$(Acctime + Dectime) < (number_of_table_points/4000)$$

If this criteria is not met, a current spikes at 4KHz will occur.

11.2.4.5 Profile Table Interpolation Motion Task

The Profile Table interpolation motion task is displayed in the following figure:



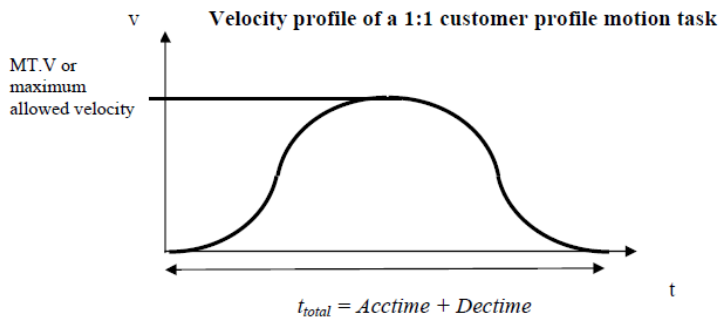
The Profile Table interpolation for a stand-alone motion task (the motion task does not automatically trigger a following motion task) can be separated in three different phases:

1. Phase I: The drive steps within a pre-calculated acceleration time through the first half of the motion profile table and reaches finally the requested target velocity of the motion task.
2. Phase II: The drive inserts a constant velocity phase and checks continuously if a brake-point has been crossed. The brake-point is naturally the target position minus the deceleration distance.
3. Phase III: The drive steps into the second half of the motion profile table and reaches finally the requested target position when the velocity becomes zero. The step into the second half of the motion profile table is a critical point and requires a symmetric table and the value of 231 at entry number_of_table_points / 2 + 1 as explained in 11.2.4.2 Motion Profile Table Restrictions for a customer table.

11.2.4.6 OneToOne interpolation motion task

The OneToOne interpolation motion task is basically very similar to the Profile Table interpolation motion task handling with just a few small differences.

1. The OneToOne customer table motion task does not step out of the table after an acceleration process and inserts a constant profile (Phase II in the section above). The OneToOne handling steps within a pre-calculated time through the whole table in one go and cover the required distance.
2. A change-on-the-fly from one motion task to another without finishing the first motion task is not possible for this mode.
3. The OneToOne profile does not use different acceleration and deceleration values. The AKD PDMM calculates the sum of the acceleration time and deceleration time and uses this total time (total τ = $MT.V/DRV.ACC + MT.V/MT.DEC$) for the motion task as explained in the following picture. In case that the acceleration + deceleration time is too small for moving a certain distance, which would lead into a too large peak-velocity, the total time will automatically be extended to the required value in order to not exceed the maximum allowed velocity to the minimum of $MT.V$ or $VL.LIMITP$ and $VL.LIMITN$.



Note that the motion task target velocity is only reached in case of a symmetric table. The velocity will be different when the profile table is non-symmetric.

11.2.4.7 Setting up a motion profile motion task

It is recommended that motion tasks be set up via the [EWV¹](#) software. The S-curve profile or the OneToOne profile may be selected via a drop-down menu.

It is also possible to select to adjust a motion task on a command line level with the help of the MT commands. There are 2 commands which are mentioned within this chapter:

- Trapezoidal, OneToOne, or Profile Table moves are selected using bits 10 and 11 of the MT.CNTL command.
- The MT.TNUM selects which of the 8 tables (0 to 7) to use for the profile. The parameter MT.TNUM will be ignored in the case that a trapezoidal motion task has been selected.

11.2.4.8 Drive reaction on impossible motion tasks

For all motion tasks, which use a motion profile table as the shape for the velocity profile, the motion task properties must be pre-calculated and it must be evaluated in advance, if a motion task can be handled without any problems or if some of the motion task parameters must be re-calculated automatically by the AKD PDMM.

An impossible motion task occurs when the user has not specified enough movement in order to accelerate to the motion task target velocity and to decelerate to velocity 0 without exceeding the distance to travel.

11.2.4.8.1 OneToOne interpolation limitations

A OneToOne interpolation motion task cannot be activated while another motion-task is running. A OneToOne interpolation motion task must start from velocity 0.

When using a OneToOne interpolation motion task the AKD PDMM pre-calculates the expected peak-velocity and checks if the velocity exceeds the minimum of the MT.V, VL.LIMITP (→ p. 748) and VL.LIMITN (→ p. 746).

The expected peak-velocity according to the figure above can be calculated via using the following formula:

$$V_{PeakExpected} = \frac{2 \cdot \text{Distance_to_travel}}{t_{total}}$$

The 'distance to travel' is defined in the motion task settings MT.P and MT.CNTL. In case that VPeakExpected exceeds the minimum of the MT.V, VL.LIMITP or VL.LIMITN setting, the AKD PDMM re-calculates the total so that VPeakExpected does not exceed the velocity limitations.

¹Embedded Workbench Views

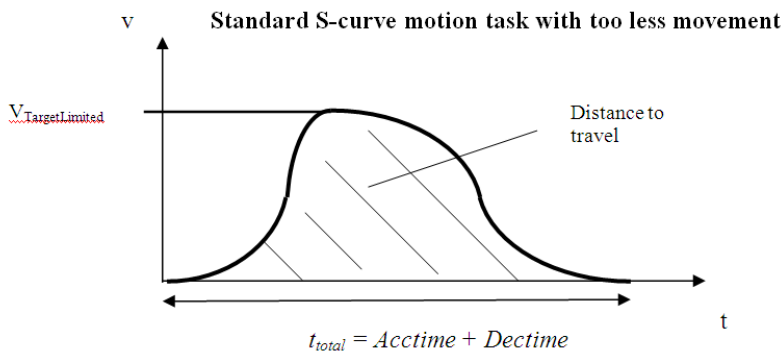
The AKD PDMM accelerates and decelerates within the same time in case of a OneToOne profile and therefore different settings for MT.ACC and MT.DEC are not considered.

11.2.4.9 Profile Table interpolation limitations

11.2.4.9.1 Starting from velocity 0 without change-on-the-fly to a following motion task

If there is not enough ‘distance to travel’ selected by the user in order to accelerate to the target velocity via the selected acceleration (internally converted to acceleration time) and deceleration (internally converted to deceleration time), the AKD PDMM lowers the target velocity automatically to $V_{TargetLimited}$ and accelerates within the selected acceleration time to the limited target velocity and decelerates afterward with the selected deceleration time to velocity 0.

The shape of the velocity profile will look like the following pictures with the assumption, that MT.ACC and MT.DEC have different values.



11.2.4.9.2 During a change on the fly condition

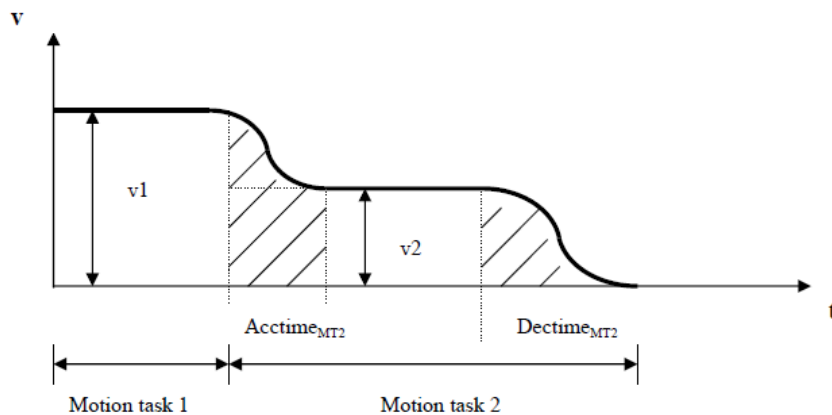
There are 2 different kinds of considerations within the AKD PDMM firmware for a change-on-the-fly condition.

- A change on the fly in the same direction (the target velocity of the previous and the following motion task have the same algebraic sign).
- A change on the fly in the opposite direction (the target velocity of the previous and the following motion task have a different algebraic sign).

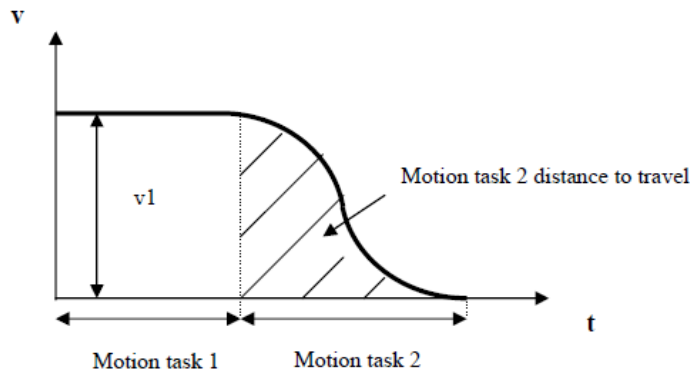
Since the shape of a customer table is unknown to the AKD PDMM, the drive verifies in advance the validity of the motion task with the assumption of a symmetric motion profile table.

11.2.4.9.3 Movement to the same direction

The following figure displays a movement in the same direction, in this case in a positive direction.

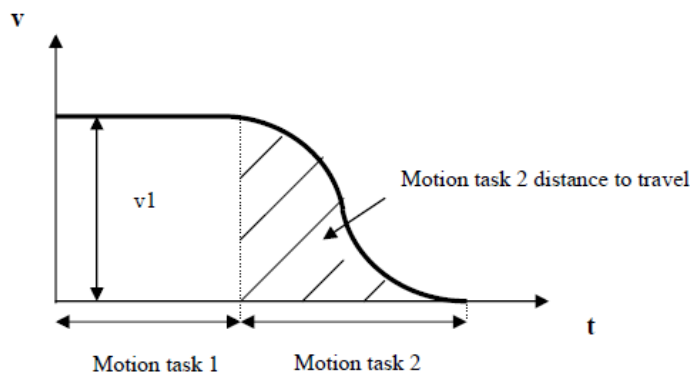
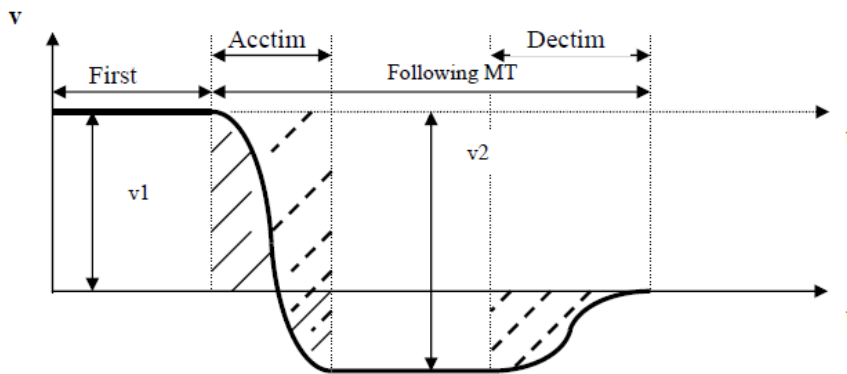


In case that the distance to the target position of the motion task 2 is smaller than $dist_{min}$, the AKD PDMM generates a profile as shown in the next figure.



11.2.4.9.4 Movement in different directions

The switch on the fly from a positive velocity to a negative velocity is described in the next figure.



It is not possible to pre-calculate exactly the area, which is marked with solid lines of the following motion task since the shape of the motion profile table is unknown to the drive. This means that it is not possible to identify the movement in positive and negative direction during a change on the fly from v_1 to v_2 . A criterion that a change on the fly will be executed by the drive is, if the total movement in negative direction of the following MT is larger than the area, which is marked with dashed lines. In this case it is ensured, that there will be definitely enough total movement of the MT in negative direction, because the motor moves during the acceleration from v_1 to v_2 also a bit in positive direction. The magnitude of v_2 is in this case the 'target velocity of MT1' + 'target velocity of MT2.'

The drive behaves as follows in case that the hatched area is smaller than the distance to travel

negative direction:

1. The drive stops the first motion task with the assigned deceleration ramp.
2. Afterwards the following motion task is triggered automatically by the drive starting from velocity 0.

12 Tuning Your System

| | |
|---|------------|
| 12.1 Introduction | 103 |
| 12.2 Slider Tuning | 103 |
| 12.3 Using the Performance Servo Tuner | 104 |
| 12.4 Tuning Guide | 132 |
| 12.5 Cogging Compensation | 149 |

12.1 Introduction

Most servo systems require some level of tuning (setting up the desired response of the system usually with the load attached). This can be done through several methods available within [EWW¹](#).



Slider Tuning - Slider tuning offers a very simplistic approach to tuning and can get you underway quickly. With this method, just the proportional and integral gains are adjusted based on the desired bandwidth you wish to achieve. You can take into account your load inertia if it is known. None of the Bi-quad filters are impacted by the slider tuning.



Performance Servo Tuner - This is an easy way to get a more sophisticated tuning accomplished. The details of how the PST works are included in the advanced section of this subject. However, the PST approach is a simple one-button solution that lets the system configure all of the tuning parameters for you. The PST is a robust solution to get your system tuned and ready to operate in a wide variety of mechanical configurations and loads.

Manual Tuning - Certain applications may require manual tuning where you set the gains and filters based on the specific performance you are looking for. You may also need to "tweak" the tuning that was set up in either the slider or the PST section to optimize your application performance.

12.2 Slider Tuning

This view allows you to vary the tuning of your drive using the slider.



Slider Tuning

[? Learn more about this topic](#)

Select the stiffness you would like.

Do you know your load inertia?

Load inertia: kgcm²

Select desired bandwidth

For very heavily loaded systems.

Default. For heavily loaded or softly coupled systems.

For unloaded and lightly loaded systems.

Bandwidth: Adjust the slider to your desired stiffness

1 300

12.2.1 Load Inertia

¹Embedded Workbench Views

If you know your load inertia (LOAD.INERTIA (→ p. 589)), enter it in kgm^2 for rotary motors, and kg for linear motors. If no load inertia is entered, $\underline{\text{EWV}}^1$ will assume a 1:1 ratio between your load inertia and motor inertia.

12.2.2 Gentle, Medium, and Stiff

These buttons select three of the most common bandwidths:

- Gentle works in all but the most challenging situations.
- Medium is the default and works in most situations.
- Stiff works for unloaded motors.

12.2.3 The Slider

As you drag the slider to the right, the stiffness increases. In many situations, you cannot drag the slider fully to the right side because the system will become unstable.

12.3 Using the Performance Servo Tuner

12.3.0.1 Overview

The Performance Servo Tuner (PST) tunes your system quickly and easily. The advanced technology in the PST achieves high performance and stability for both simple and complicated loads. It also collects frequency response data (bode plot) that can be used for advanced analysis. The PST can work in a “one button” mode that requires no customization. Settings can be also changed for an array of system requirements (Using the Performance Servo Tuner: Advanced (→ p. 108)).

NOTE

The PST will **not** work reliably on a vertical axis, since the motor may rest against an end stop. In this case, the inertia will not always be correctly identified.

NOTE

The PST is not available when FB1.SELECT = 12 (Halls Only). The resolution of this feedback is too low for a quality autotune. Due to heavy quantization, large current spikes occur at hall transition points, it is advisable to tune the observer to a low frequency (like 10Hz) and set the observer gain appropriately to achieve the best operating results.

The default PST settings tune systems quickly for good servo performance. If your tuning is not acceptable after running PST, try adjusting the following settings.

- The most common unacceptable tuning result is that the servo bandwidth is too high (very stiff and audibly noisy). To avoid this, try setting the gain margin to 16db and the phase margin to 55deg. This will result in a lower bandwidth and therefore less audible noise. It may also be softer on the system mechanics, but still offer exceptional performance. Use the motion scope to check settling times and overshoots on a typical move to measure performance.
- The PST collects data on the system as it runs. It is important for the PST to collect a robust data set. Try increasing FFT points to 16,384 and number of points to 250,000. This increases the data collection accuracy at low frequency excitations, which can improve the resulting tuning performance. Increasing FFT points and number of points also increases the data collection time. Timing requirements depend on the processing power of the PC running AKD Workbench. If the tuning is not complete after an extreme period of time, cancel the tuning and reduce the points. If possible, try running on a more powerful PC.

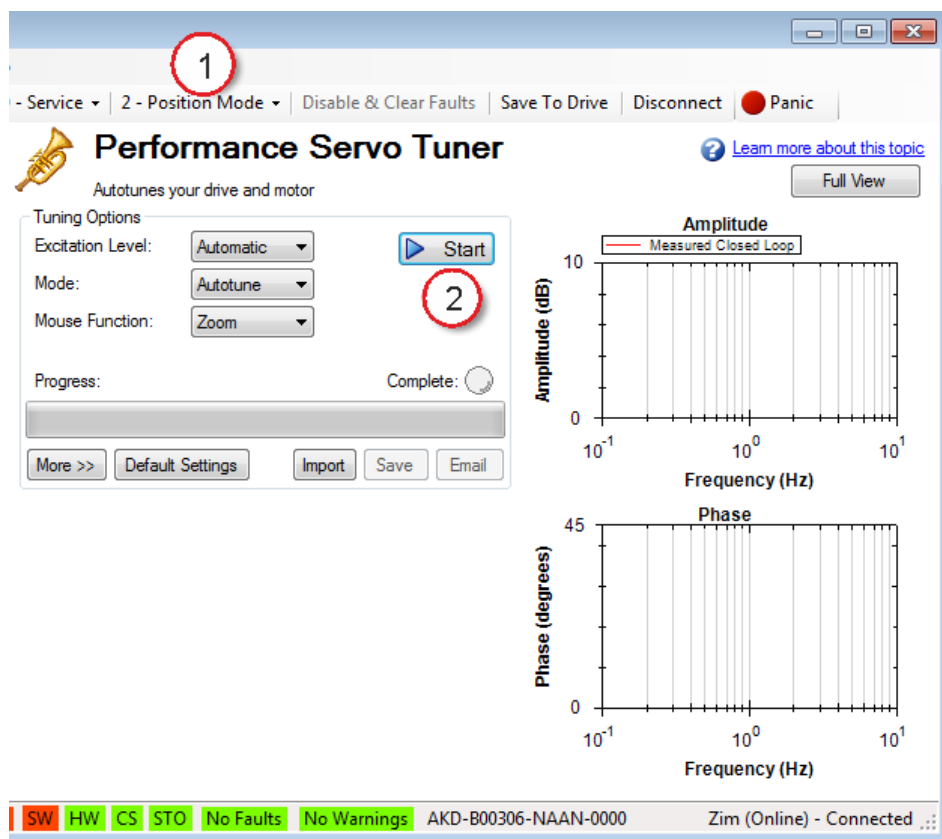
Look through the advanced tuning section for more information. Complex systems may require more adjustment.

¹Embedded Workbench Views

12.3.0.2 Using the PST

The PST defaults to a “one button” mode, in which the PST is completely automatic after pushing the start button. In the **Settings** select your desired operation mode, navigate to the **Performance Servo Tuner** view, and then tune your system as follows:

1. Select whether you would like the drive to be tuned in **1-Velocity** or **2-Position** Mode. If the drive is in torque mode, the PST will tune in position mode by default. This is set by using the tool bar and changing the mode to either velocity or position. The drive must be disabled in order to change the operation mode.
2. Click **Start**.



The PST will then perform several tests and display results as shown below. The progress bar (1) shows the relative progress of the PST, so you can estimate when the tuning will be finished. When the tuning is complete, the green **Complete** LED (2) illuminates, and a Bode plot (3) is displayed showing the frequency response of the tuned system.

Performance Servo Tuner
Autotunes your drive and motor

[Learn more about this topic](#) Full View

Tuning Options

Excitation Level: ▶ Start

Mode:

Mouse Function: 2

Progress: Complete 1 Complete:

Autotuner | Measurement | Plots | Cursors | Note

Current Amplitude: 0.095 Measurement:

Velocity Amplitude: 3.063 Injection Point:

Smoothing %: 1.000 Excitation Type:

FFT Points: 4096 Velocity Max: 1000.00 (RPM)

Number Points: 60000 Excite Gap: 2

Amplitude

— Measured Closed Loop 3

Amplitude (dB) vs Frequency (Hz)

Phase

Phase (degrees) vs Frequency (Hz)

12.3.0.2.1 Saving and Emailing Bode Plots

To save screenshots and raw data of a Bode plot, click on either **Save** (1), or **Email** (2).

Tuning Options

Excitation Level: ▶ Start

Mode:

Mouse Function:

Progress: Complete Complete:

1 2

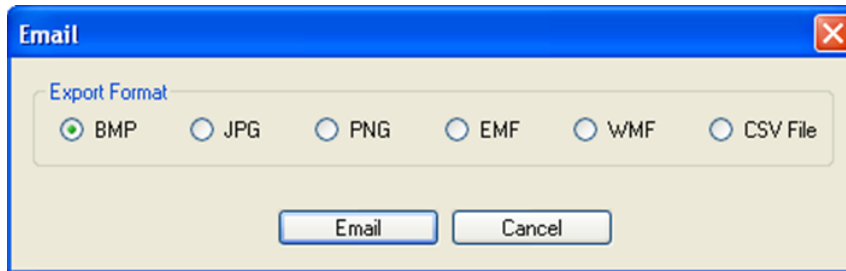
Clicking **Save** opens a prompt to select how to save the screenshot or data. Selecting BMP, JPG, PNG, EMF, or WMF saves the Bode plot as an image. Choosing CSV saves the raw data that is currently plotted as a comma delimited file. Click **Save** to save the file to your hard drive in the desired format.

Save Image/CSV

Export Format

BMP JPG PNG EMF WMF CSV File

Clicking **Email** opens a similar prompt. Select the file format in which you wish to save the image or raw data, and an email will be created for you with the file automatically attached for your convenience. Click **Email** to create the email with the selected file attachment.



12.3.0.2.2 Importing a Frequency Response

If you have previously saved a frequency response measurement into a CSV file, it can be imported for later viewing. Click the **Import** button, and browse to your saved CSV file. You can import while in offline mode for convenience. Importing a frequency response is useful for off-site developers to analyze a machine tool.

12.3.0.3 Measurement Options

By default, the PST determines the excitation level automatically and autotunes the drive and motor.

The PST also allows you to enter a manual excitation level or to take only Bode measurements (without auto-tuning the system).

12.3.0.3.1 Using Manual Excitation Levels

By default, the PST is set to use the automatic excitation level. To obtain the automatic excitation level, the PST runs a friction test at the beginning to determine how much excitation is needed to break friction and get an accurate measurement.

To change this excitation level, click on the **Excitation Level** drop-down box (1), and select **Manual**. Then enter a new **Current Amplitude** (2) in amps.

Note: If the **Injection Point** is set to **Current**, then the **Current Amplitude** box will be enabled to enter an excitation level; if the **Injection Point** is set to **Velocity**, the **Velocity Amplitude** box will be enabled to enter an excitation level.

Performance Servo Tuner
Autotunes your drive and motor

[Learn more about this topic](#) Full View

Tuning Options
 Excitation Level: Manual 1 Start
 Mode: Autotune
 Mouse Function: Zoom
 Progress: Complete Complete:
 Less << Default Settings Import Save Email

Autotuner 2 **Measur** Plots Cursors Note
 Current Amplitude: 0.095 Measurement: Closed Loop
 Velocity Amplitude: 3.063 Injection Point: Current
 Smoothing %: 1.000 Excitation Type: PRB
 FFT Points: 4096 Velocity Max: 1000.00 (RPM)
 Number Points: 60000 Excite Gap: 2

Amplitude
 Measured Closed Loop
 Amplitude (dB) vs Frequency (Hz)

Phase
 Phase (degrees) vs Frequency (Hz)

SW HW CS STO No Faults No Warnings AKD-B00306-NAAN-0000 Zim (Onl... ..)

12.3.0.4 Taking a Bode Measurement without the PST

You may wish to take only the frequency response of a system, rather than using the PST. To take a frequency response measurement without the PST, click on the **Mode** drop-down box (1) and select **Bode Plot**, then click **Start** (2).

Tuning Options
 Excitation Level: Manual Start
 Mode: Bode Plot 1 2
 Mouse Function: Zoom
 Progress: Complete Complete:
 Less << Default Settings Import Save Email

12.3.1 Using the Performance Servo Tuner: Advanced

The Performance Servo Tuner (PST) can be set up to use specific modes or limits in tuning to provide tuning in ways you can control, while still taking advantage of the PST's ability to make decisions quickly and effectively for you.

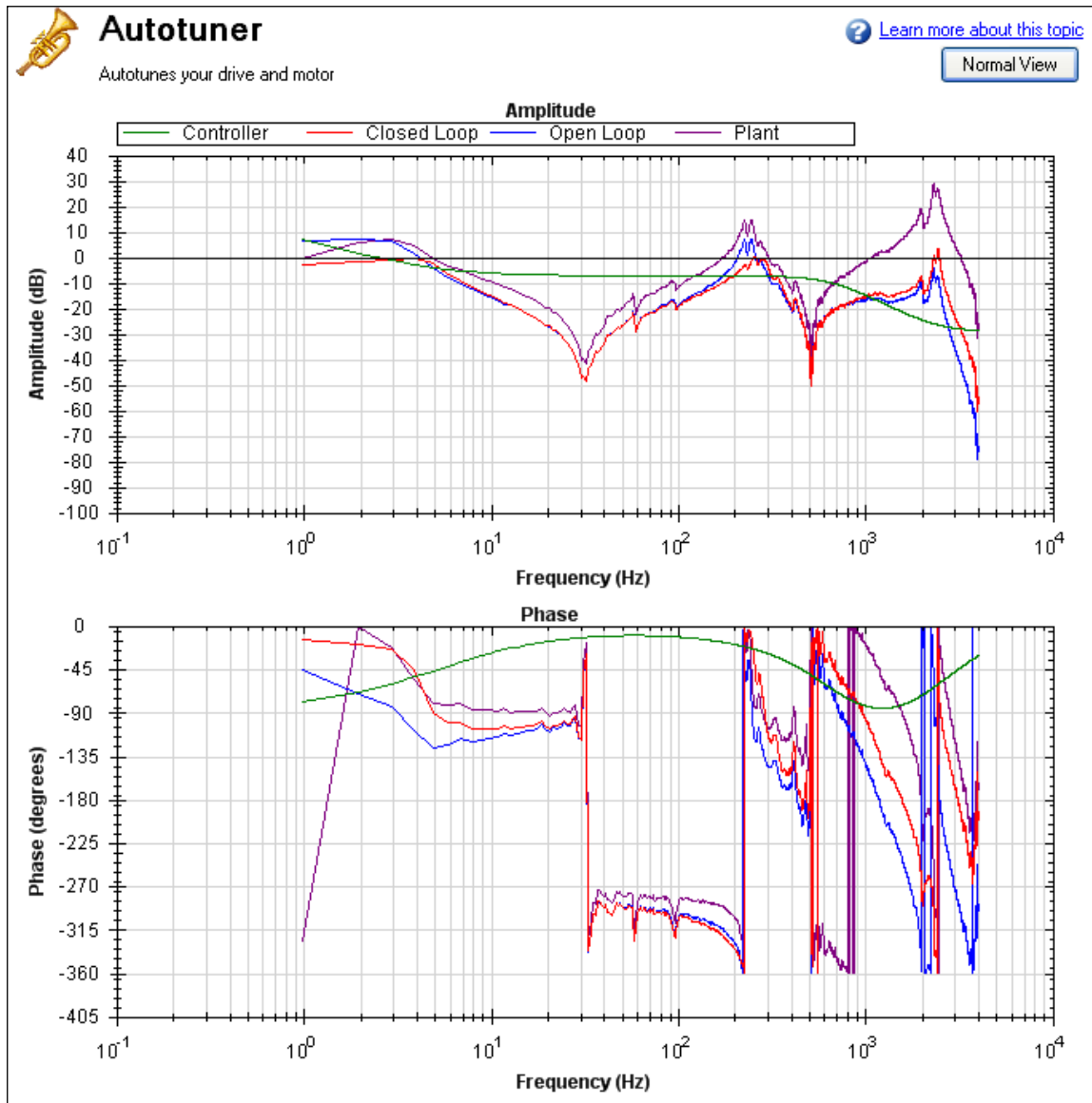
To use the advanced modes of the PST, click the **More** button to display the additional features for advanced autotuning:

12.3.1.1 Typical Cases for Advanced PST Use

Tuning Systems with Low-Frequency Resonances

Systems with low-frequency resonances are challenging because low frequency data is difficult to measure. While the PST can tune these systems, you can expect lower system performance. If your system has a first

anti-resonance of 30 Hz (pictured below), you can expect approximately 15 Hz (half the frequency of the first anti-resonance) of closed loop bandwidth.

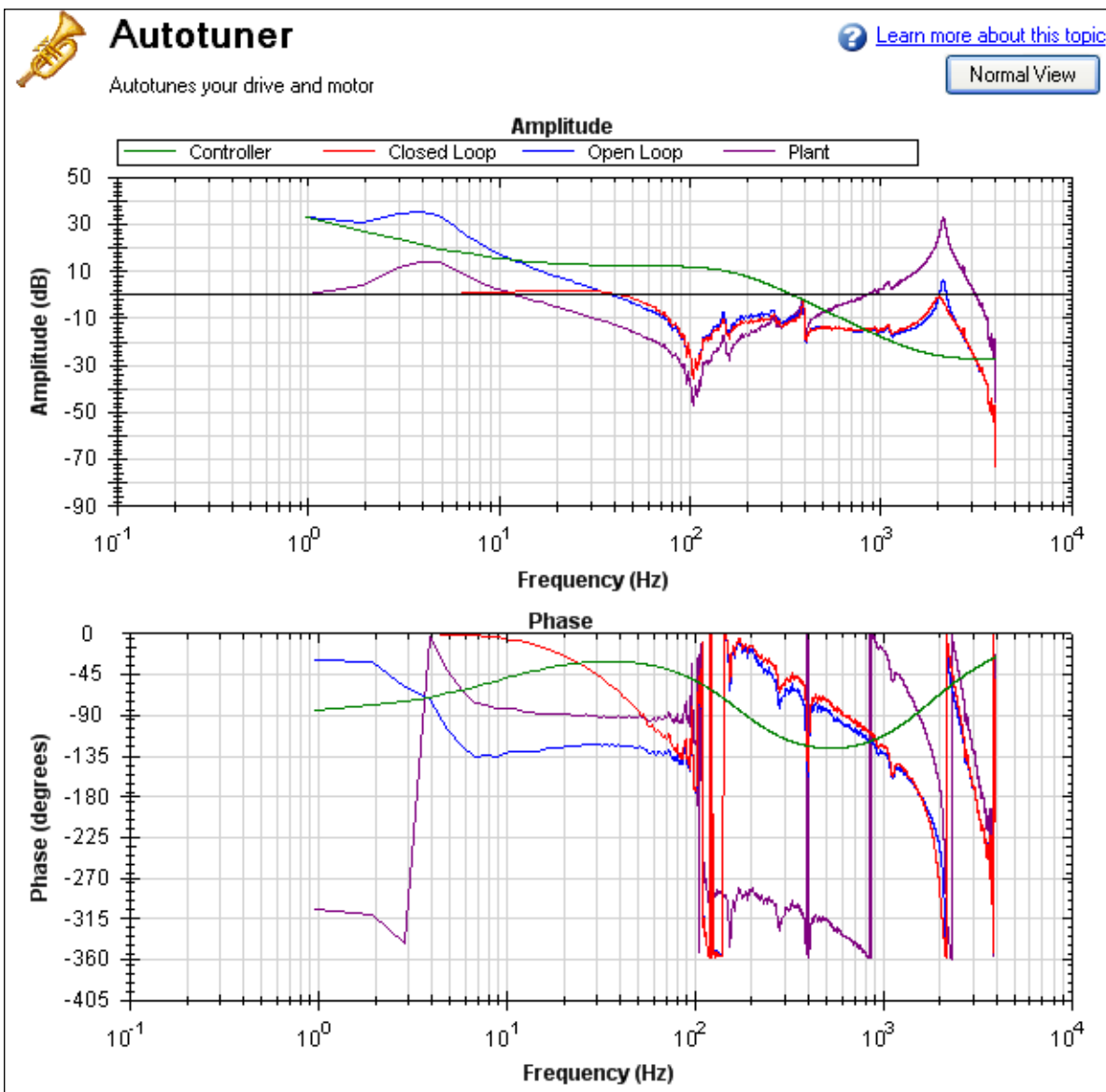


In addition, in order to accurately measure the low frequency resonances, the fast Fourier transform (FFT) resolution must be sufficiently fine to accurately measure the low-frequency resonance. A good place to start is to have an FFT resolution of 1/10 of the frequency of the lowest anti-node. In the case shown above, an anti-resonance of 30 Hz is present, so the resolution should be approximately 3 Hz FFT resolution. The PST can function with the resonance if it is accurately measured, as shown below. To adjust the FFT resolution, adjust **FFT Points** in the **Recording Options** tab as needed.

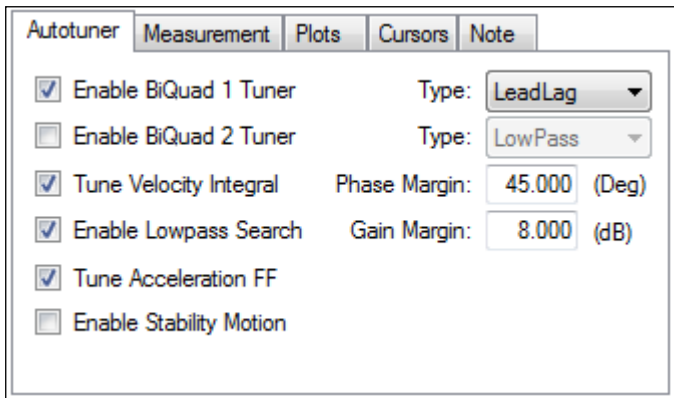
| Autotuner | Measurement | Plots | Cursors | Note |
|---------------------|-------------|------------------|---------------|------|
| Current Amplitude: | 0.095 | Measurement: | Closed Loop | |
| Velocity Amplitude: | 3.063 | Injection Point: | Current | |
| Smoothing %: | 1.000 | Excitation Type: | PRB | |
| FFT Points: | 4096 | Velocity Max: | 1000.00 (RPM) | |
| Number Points: | 60000 | Excite Gap: | 2 | |

Tuning Systems with High-Frequency Resonances

Some systems have resonances at very high frequencies (greater than 1 kHz). When the resonance is this large, it can prove a challenge in tuning, because these systems generate high noise levels that are often audible. An example of a large resonance is shown below. This example is from a steel flywheel mounted to an AKM 22E motor. The source of the resonance is the spring mass relationship between the motor rotor, shaft, and flywheel.

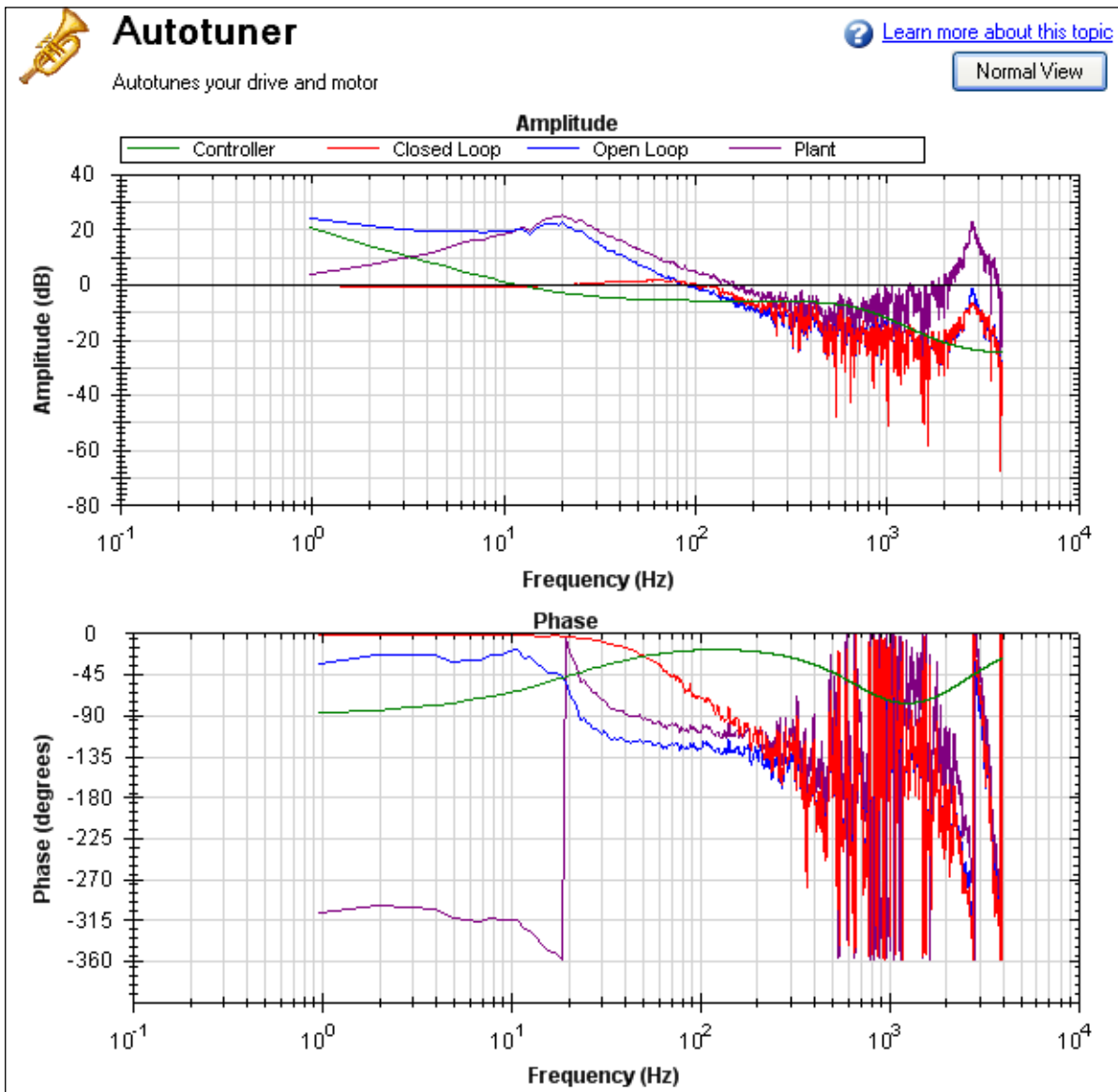


One way to resolve this problem is to use a low-pass filter in the feedback path. To use this filter, simply check the **Enable Lowpass Search** in the PST, which is the default behavior.



Tuning systems with noisy frequency responses

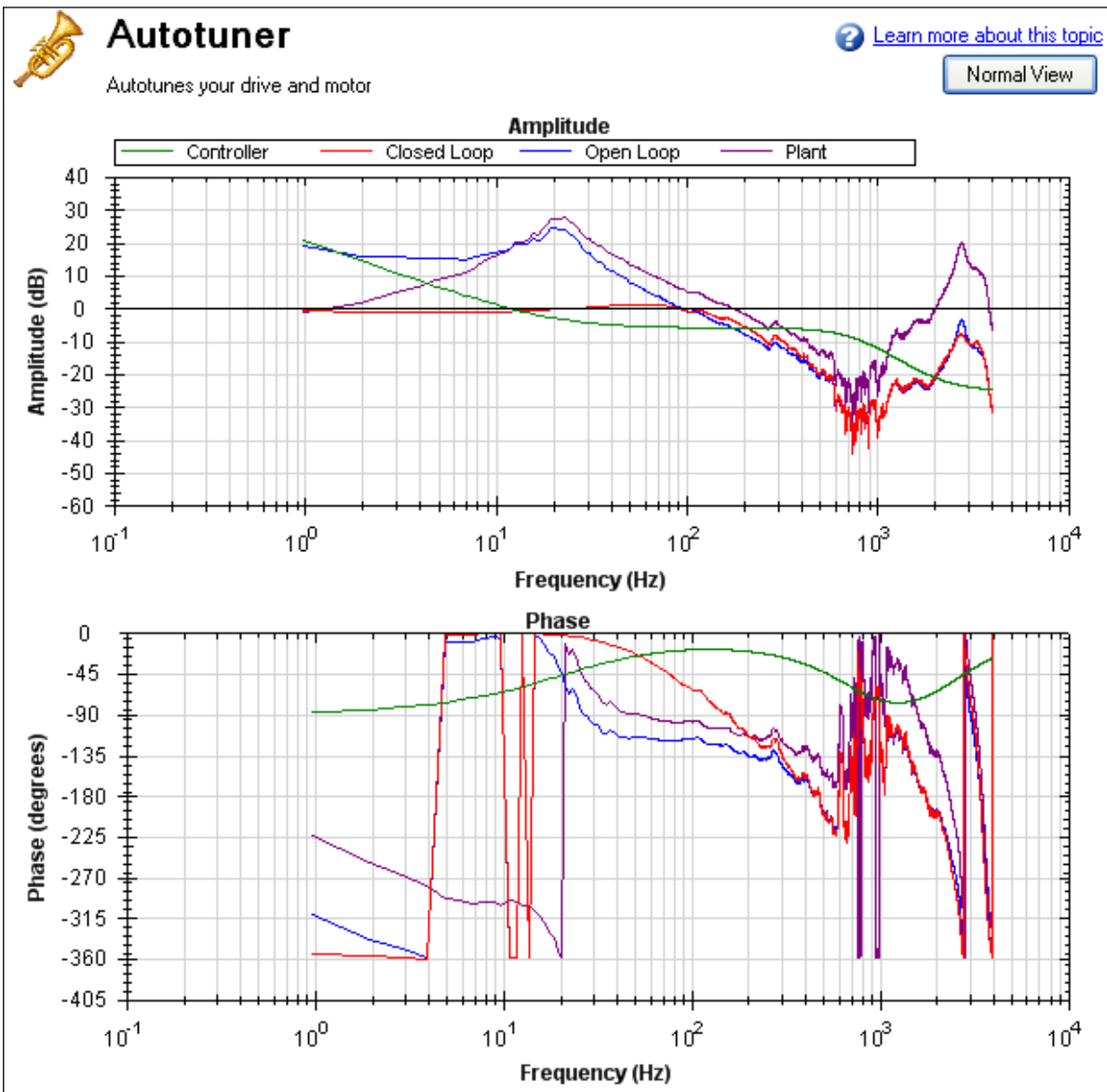
When using a motor with a low-resolution incremental encoder or resolver, the high frequency response may be noisy. Below is a Bode plot created after autotuning of an incremental encoder with 8,192 counts per revolution.



To make the Bode plot easier to read, increase the smoothing factor (1) in the advanced **Measurement Options**.

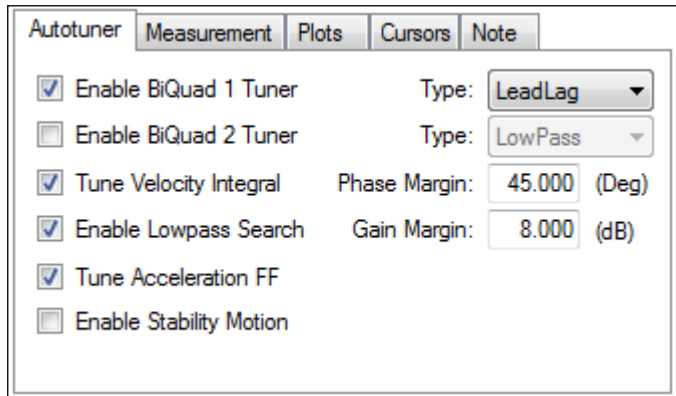
| Autotuner | Measurement | Plots | Cursors | Note |
|---------------------|-------------|------------------|---------------|------|
| Current Amplitude: | 0.095 | Measurement: | Closed Loop | |
| Velocity Amplitude: | 3.063 | Injection Point: | Current | |
| Smoothing %: | 4.000 | Excitation Type: | PRB | |
| FFT Points: | 4096 | Velocity Max: | 1000.00 (RPM) | |
| Number Points: | 60000 | Excite Gap: | 2 | |

After increasing the smoothing percentage, the Bode plot traces become cleaner and easier to read:



12.3.1.2 PST Options

When you click **More** in the PST view, the following options are displayed:



| Option | Status | Value | Unit |
|-------------------------|-----------|---------|------|
| Enable BiQuad 1 Tuner | Checked | LeadLag | Type |
| Enable BiQuad 2 Tuner | Unchecked | LowPass | Type |
| Tune Velocity Integral | Checked | 45.000 | Deg |
| Enable Lowpass Search | Checked | 8.000 | dB |
| Tune Acceleration FF | Checked | | |
| Enable Stability Motion | Unchecked | | |

Enable BiQuad 1 Tuner

Check this box to use the first anti-resonance filter in the forward path (AR1). You can specify the type of filter to use in the **Type** box to the right of **Enable BiQuad 1 Tuner**.

Enable BiQuad 2 Tuner

Check this box to use the second anti-resonance filter in the forward path (AR2). You can specify the type of filter to use in the **Type** box to the right of **Enable BiQuad 2 Tuner**. Enabling this option may significantly slow your computer during this operation.

Biquad Type

For Biquad 1 and 2, you can choose what type of filter to implement. The four options are:

1. **LeadLag**: The LeadLag filter is the default, and will work for most servo systems.
2. **Lowpass**: A Lowpass filter requires the least amount of processing time. The PST will place the low-pass to get the maximum bandwidth possible.
3. **Resonator**: The Resonator filter is like a Notch filter with tunable bandwidth and notch depth. The Resonator takes longer to calculate than the LeadLag filter.
4. **Custom**: The Custom filter takes the longest to calculate and does not restrict the PST to a filter shape. This filter type provides excellent results, but may significantly slow your computer while the filter is calculated.

Tune Acceleration FF

This box turns on and off the acceleration feedforward tuner. If this box is checked, the PST will measure the inertia attached to the motor shaft, and using this measurement, will calculate an appropriate acceleration feedforward and write it to the drive (IL.KACFF (→ p. 565))

Enable Stability Motion

When this checkbox is checked, after the PST has completed, the PST will command a short move in the clockwise direction, then back to its origin and monitor the motor's parameters to determine if the tuning is stable. If an instability is detected, the drive will generate Fault F133 (→ p. 179): Instability during Autotune.

Phase and Gain Margins

The PST always ensures that the tuning satisfies stability criteria that can be adjusted in units of phase margin (in degrees) and gain margin (in dB). The PST uses default values for phase and gain margin, but you can adjust these values to ensure higher stability or to allow the PST to be more aggressive by using lower gain and phase margins.

Tune Velocity Integral

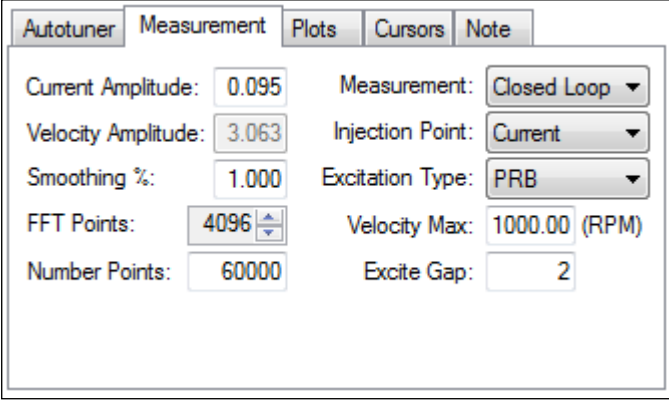
Check this box to tune VL.KI (velocity loop integral gain). If this box is unchecked, the PST will set VL.KI to zero.

Enable Low Pass Search

Check this to tune a fourth-order low pass filter in the feedback path (AR 3 and 4). If this box is unchecked, the PST will not modify the anti-resonance filters in the feedback path.

12.3.1.3 Measurement Options

The PST screen also provides options for measurements:



| Autotuner | Measurement | Plots | Cursors | Note |
|---------------------|-------------|------------------|---------------|------|
| Current Amplitude: | 0.095 | Measurement: | Closed Loop | |
| Velocity Amplitude: | 3.063 | Injection Point: | Current | |
| Smoothing %: | 1.000 | Excitation Type: | PRB | |
| FFT Points: | 4096 | Velocity Max: | 1000.00 (RPM) | |
| Number Points: | 60000 | Excite Gap: | 2 | |

Current Amplitude

This box sets the amplitude of the current used to excite the system during a current injection mode excitation. This amplitude applies to all excitation types when the **Injection Point** is set to **Current**. The **Current Amplitude** box is disabled if the **Injection Point** is set to anything else.

Velocity Amplitude

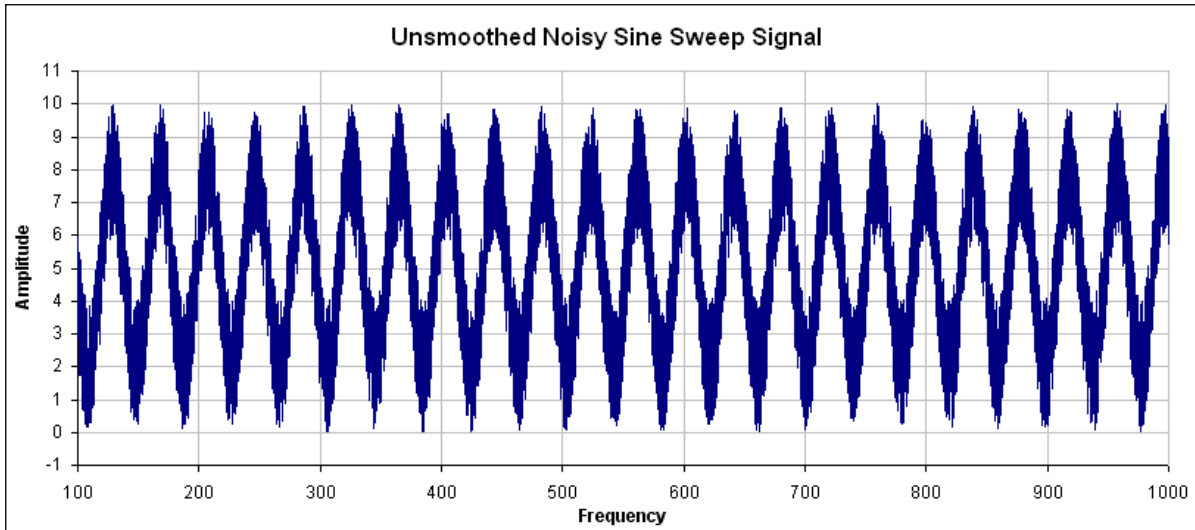
This box sets the amplitude of the velocity used to excite the system during a velocity injection mode excitation. This amplitude applies to all excitation types when the **Injection Point** is set to **Velocity**. The **Velocity Amplitude** box is disabled if the **Injection Point** is set to anything else.

Smooth %

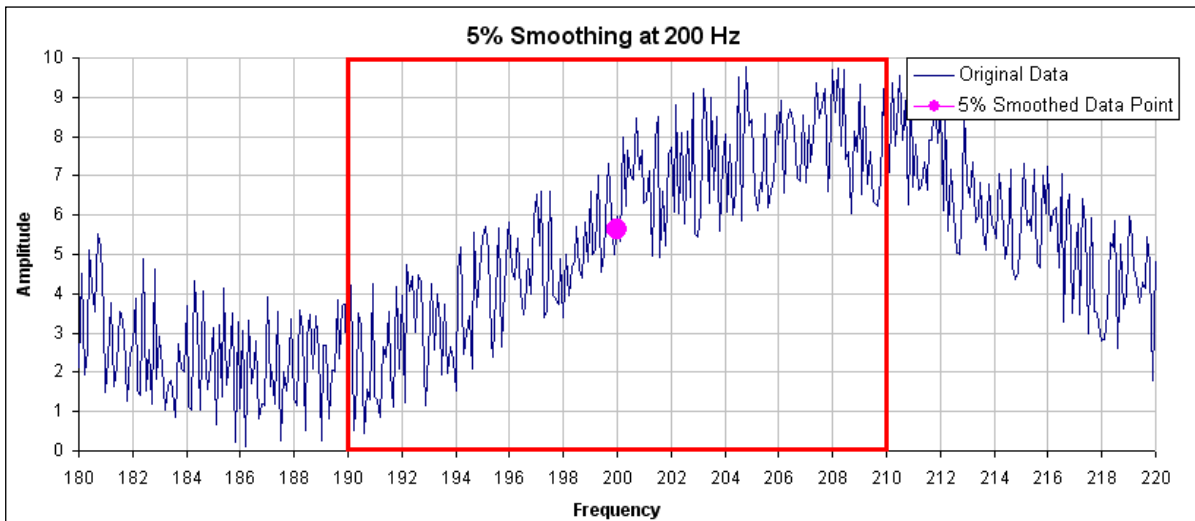
This value applies a moving average smoothing filter to the frequency response gathered during autotuning. This process reduces noise in the frequency response that can occur when making short frequency response measurements, using low resolution encoders, conducting low amplitude frequency response tests, or for other reasons. The smoothing filter iterates through each frequency on the FFT plot. For each frequency, all frequencies within the **Smooth %** range will have their magnitudes averaged.

For example, if you smooth a Bode plot with 5% smoothing, at 100 Hz, it will average all the values between 95Hz and 105Hz; when the filter gets to 1000 Hz, the filter will average all the values between 950 Hz and 1050 Hz.

As an example, assume a noisy sine sweep signal and use a 5% smoothing factor. Below is a noisy signal with a range of 100 Hz to 1000 Hz.

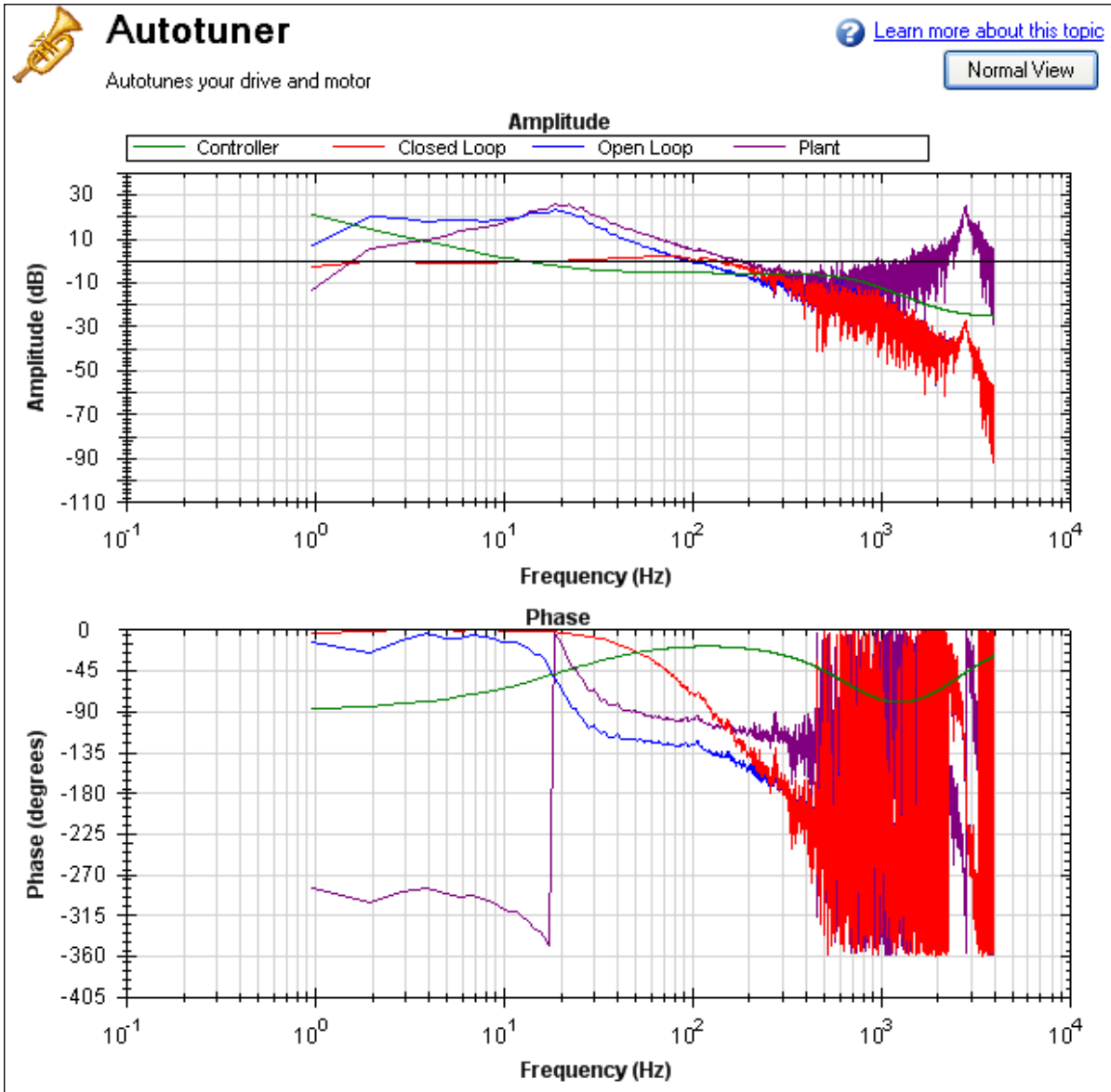


In this example, examining how the smoothing filter affects a single point shows how the smoothing filter works on a full plot. If you zoom in on 200 Hz +/- 5%, this gives a range of 190 Hz – 210Hz. The smoothing filter averages this range of values and puts the average right on 200 Hz. The figure below shows the zoomed data around 200 Hz and the averaged value of all frequencies +/- 5% (the red box illustrates the range of frequencies being smoothed).

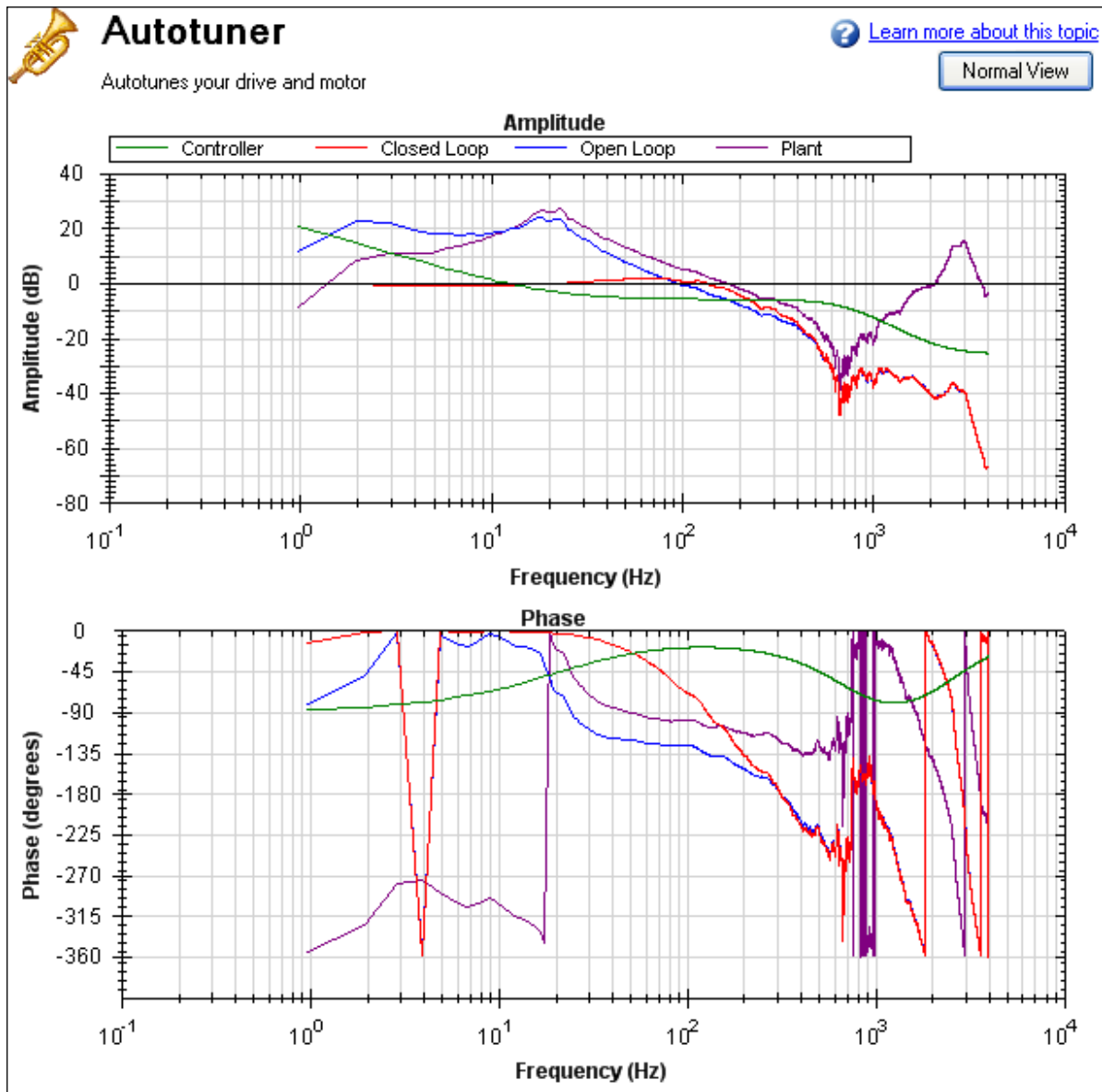


In the PST, the smoothing filter will do this analysis for every frequency point on the Bode plot. If the data is too noisy, then you can increase the smoothing percentage to smooth the noise out and see the underlying data patterns. A comparison of a system with 0.1% smoothing and 8% smoothing is shown below.

0.1% smoothing



8% Smoothing



Note: Smoothing decreases the peaks of resonances; if smoothing is too high, a resonance may be completely hidden. If the PST cannot identify a resonance due to high smoothing, the system may become unstable.

Measurement

This box sets the measurement type used during a measurement. The PST functions only if **Plant** measurement is selected; autotune does not function in other measurement modes.

- **Closed Loop** directly measures the closed loop frequency response of the servo.
- **Plant** directly measures the plant, including drive, motor, and mechanics coupled to the motor.
- **Controller** directly measures the controller response, which includes the tuning in the velocity and position loops, and anti-resonance filters 1 & 2.

Injection Point

The **Injection Point** box sets the source location of the excitation used during autotuning. **Current** mode uses a torque disturbance at the torque output. During current injection point measurements, the excitation will use the **Current Amplitude** value to set the size of the excitation.

Velocity mode uses a velocity command to excite the system. During velocity injection point measurements, the excitation will use the **Velocity Amplitude** value to set the size of the excitation.

Excitation Type

The **Excitation Type** box allows you to choose the type of excitation. Noise, pseudo random binary (PRB), and sine are the options available.

- **Noise** uses a pseudo random noise signal to excite the system. The signal varies between +/- current or velocity amplitude (depending on injection point). The signal contains a frequency spectrum that goes from a lower limit equal to:

$16,000/(\text{Excite Gap} * \text{Number Points}) \text{ Hz}$

to a higher limit equal to:

$(16,000/\text{Excite Gap}) \text{ Hz}$

The richness of the frequency spectrum comes from variance in the amplitude of the noise signal.

- **PRB** uses a pseudo random binary signal to excite the system. The signal is either + or – current or velocity amplitude (depending on the injection point). The signal contains a frequency spectrum that goes from a lower limit equal to the larger of:

$(16,000/(2^{\text{BODE.PRBDDEPTH}} (\rightarrow \text{p. 293}) * \text{Excite Gap}))$ or $16,000/(\text{Excite Gap} * \text{Number Points}) \text{ Hz}$

to a higher limit equal to:

$(16,000/\text{Excite Gap}) \text{ Hz}$

BODE.PRBDDEPTH (\rightarrow p. 293) is set to 19 by the PST. The richness of the frequency spectrum comes from variance in the phase of the signal, not the amplitude.

- **Sine** requires that you specify the start frequency, end frequency, and frequency step size. The sine sweep takes significantly longer than a noise or PRB measurement, but is often cleaner. Be careful when selecting a step size: too large of a step size may miss important resonances, and too small of a step size increases measurement time.

FFT Points

The **FFT Points** box is only visible and applicable in noise and PRB measurements. **FFT Points** sets the resolution of the FFT's measurement. The frequency resolution is equal to

$16,000/(\text{Excite Gap} * \text{FFT Points})$

By increasing **FFT Points**, the resolution becomes finer, but noise in the frequency response increases.

Excite Gap

The **Excite Gap** box is only visible and applicable in noise and PRB measurements. This box sets how frequently the test excitation is updated. The excite gap minimum value is 1; this value is normally set to 2 for autotuning. The excite rate is 16,000/gap. You can limit high frequency excitation by increasing the **Excite Gap** value.

Number Points

The **Number Points** box is only visible and applicable in noise and PRB measurements. This box sets the length of recording while measuring the frequency response of the system. The measurement length is:

$\text{Number Points} * \text{Excite Gap} / 16,000 \text{ seconds}$

Velocity Max

The **Velocity Max** box allows the user to specify the maximum velocity the motor should be able to move while performing excitation. This box is not in effect for normal drive operation; it is only visible during the PST excitation phases. This value is implemented as soon as the PST begins, and as soon as the PST is finished, the previous overspeed threshold (VL.THRESH (\rightarrow p. 753)) is restored.

| Autotuner | Measurement | Plots | Cursors | Note |
|---------------------|-------------|------------------|---------------|------|
| Current Amplitude: | 0.095 | Measurement: | Closed Loop ▼ | |
| Velocity Amplitude: | 3.063 | Injection Point: | Current ▼ | |
| Smoothing %: | 1.000 | Excitation Type: | Sine ▼ | |
| Start Sine Freq: | 1.000 | Velocity Max: | 1000.00 (RPM) | |
| End Sine Freq: | 8000.0 | Turn Sine On | | |
| Sine Step %: | 6.000 | | | |
| Current Freq: | 0 | | | |

If **Excitation Type** box is set to **Sine**, different configuration options become available.

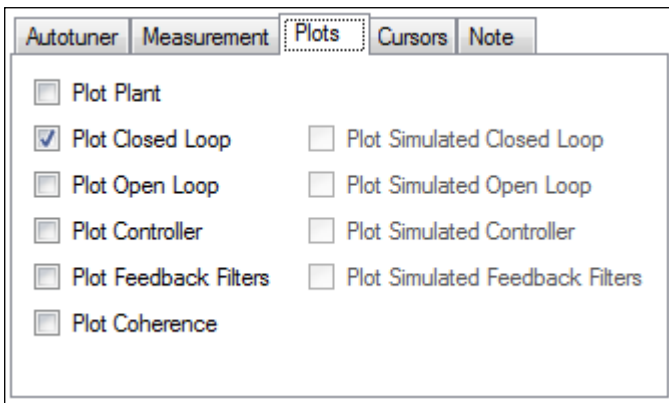
- **Start Sine Freq:** The Sine sweep test will begin at this frequency. The start frequency must be greater than zero and less than the end sine frequency. **Start Sine Freq** is only visible and applicable to Sine measurements.
- **End Sine Freq:** The Sine sweep test will end at this frequency. The end frequency must be less than or equal to 8,000, and more than the sine start frequency. **End Sine Freq** is only visible and applicable in Sine measurements
- **Sine Step %:** This box sets the sine step size. The sine sweep is discrete, not continuous. Each frequency is a multiple of the previous. For example, if the first frequency was 1 and the step size was 6%, the second frequency would be $1 * 1.06 = 1.06$ Hz, the third frequency would be $1.06 * 1.06 = 1.12$ Hz. This continues until the current frequency exceeds the **End Sine Frequency** value. **Sine Step %** is only visible and applicable in Sine measurements
- **Current Freq:** This field displays the current frequency of the sine sweep. **Current Freq** is only visible and applicable in Sine measurements
- **Turn Sine On:** This button allows the user to excite the system at a single sine frequency. When this button is pressed, it grays out boxes that do not apply. You may change the sine frequency and amplitude. To stop the sine excitation, click **Turn Sine Off**. **Turn Sine On** is only visible and applicable in Sine measurements.

| Autotuner | Measurement | Plots | Cursors | Note |
|---------------------|-------------|------------------|---------------|------|
| Current Amplitude: | 0.095 | Measurement: | Closed Loop ▼ | |
| Velocity Amplitude: | 3.063 | Injection Point: | Current ▼ | |
| Smoothing %: | 1.000 | Excitation Type: | Sine ▼ | |
| Start Sine Freq: | 1.000 | Velocity Max: | 1000.00 (RPM) | |
| End Sine Freq: | 8000.0 | Turn Sine On | | |
| Sine Step %: | 6.000 | | | |
| Current Freq: | 0 | | | |

NOTE

When the sine excitation is used on low resolution encoders, high frequency excitation may cause less than 1 count of encoder movement. If this occurs, no movement is detected on the motor for that excitation frequency. If this occurs, a data point for that frequency will not be plotted, as this results in a calculation of 0dB for gain and -infinity for phase.

12.3.1.4 Plot Options



By default, only the measured closed loop plot is selected. You can control which of these responses are displayed on the Bode plot by checking or unchecking the **Plot Plant**, **Plot ClosedLoop**, **Plot Open Loop**, **Plot Controller**, and **Plot Coherence** checkboxes shown. The options **Plot Simulated Closed Loop**, **Plot Simulated Open Loop**, **Plot Simulated Controller**, and **Plot Simulated Feedback Filters** are only available in Bode plot mode, not PST mode.

Coherence

The coherence option is only available for noise and PRB measurements; it is not available for Sine excitation measurements.

Coherence is an indicator of how accurate your data is. For example, 0 dB (1 in linear numbers) means you have perfect coherence. Another way to think of this concept is that for one unit of input, you get one unit of output. Coherence is calculated as follows:

$$Coherence = \frac{(P_{xy} \times P_{xy}^*)^2}{(P_{xx} \times P_{xx}^*) + (P_{yy} \times P_{yy}^*)}$$

where:

P_{xx} = Power Spectral Density of Input signal

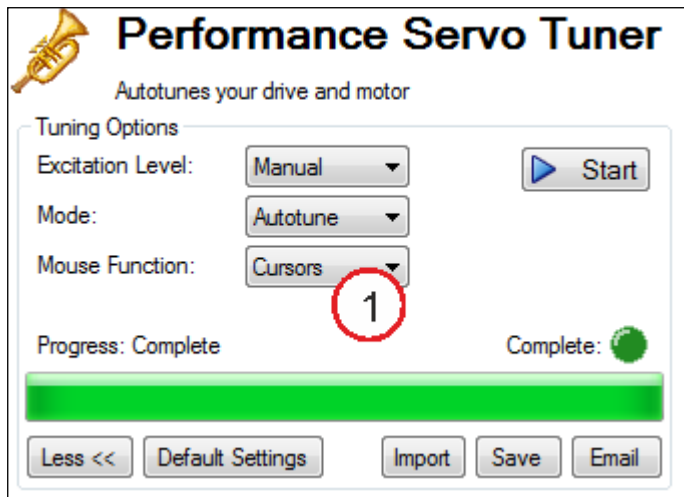
P_{yy} = Power Spectral Density of Output signal

P_{xy} = Cross Spectral Density of Input and Output

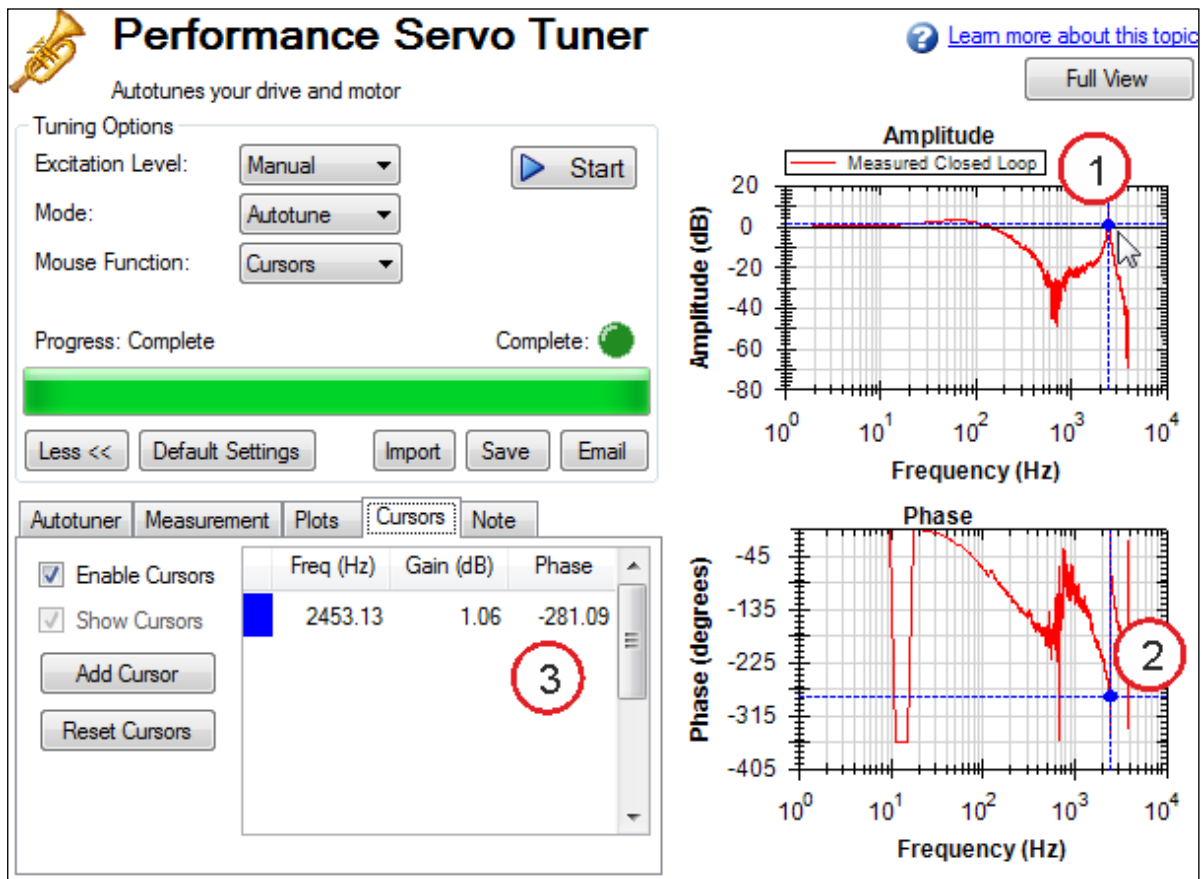
* designates complex conjugate

Cursors

Enabling cursors allows you to note specific points of interest on the Bode plot and create a table of reference points in the summary table. To enable cursors, choose Cursors from the Mouse Function drop-down(1).



To move the cursor, move your mouse over the cursor in either the Amplitude (1), or Phase (2) plots, click and hold the left mouse button, and drag the cursor to a new location. Notice as you drag the mouse, the Frequency, Gain and Phase change in the summary window (3).



To add more cursors, click **Add Cursor**; you can add 10 cursors to the Bode Plot. When selecting a cursor, the cursor closest to the mouse will be selected. While dragging the cursor, the cursor will snap to the closest trace on the plot.

When cursors are enabled, zoom functions on the graph are disabled. To re-enable zooming, switch the Mouse Function to Zoom.

The dotted crosshair lines are only drawn for the active cursor selected; to remove all cursors from the screen, but retain their position, uncheck **Show Cursors**. To reset all cursors, click **Reset Cursors**.

Note: If a CSV file is saved or emailed after placing a cursor on the Bode plot, a cursor summary is included in the CSV raw data.

12.3.1.5 Resizing Bode Plots

In the PST view, the **Full View** and **Normal View** button (1) in the upper right of the window allows you to see the Bode Plot in greater or less detail. When viewing the Bode Plot in full view, the PST settings are hidden behind the Bode Plot. To access the PST settings, click the **Normal View** button in the upper right of the window.

Simple measurement normal view

Performance Servo Tuner
Autotunes your drive and motor

[Learn more about this topic](#) Full View

Tuning Options

Excitation Level: ▶ Start

Mode:

Mouse Function:

Progress: Complete Complete:

Measurement | Autotuner | Plots | Cursors | Note

Current Amplitude: Measurement:

Velocity Amplitude: Injection Point:

Smoothing %: Excitation Type:

FFT Points: Velocity Max:

Number Points: Excite Gap:

Amplitude

— Measured Closed Loop

Amplitude (dB) vs Frequency (Hz)

Phase

Phase (degrees) vs Frequency (Hz)

Simple measurement full view

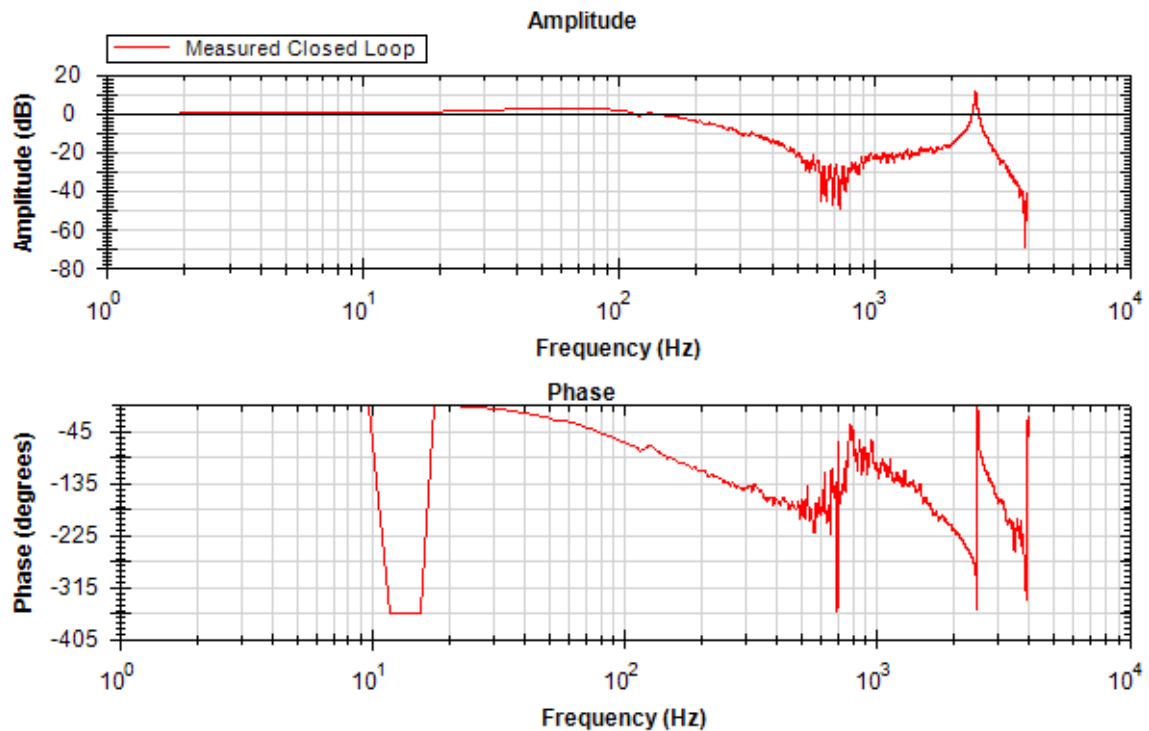


Performance Servo Tuner

Autotunes your drive and motor

[Learn more about this topic](#)

Normal View



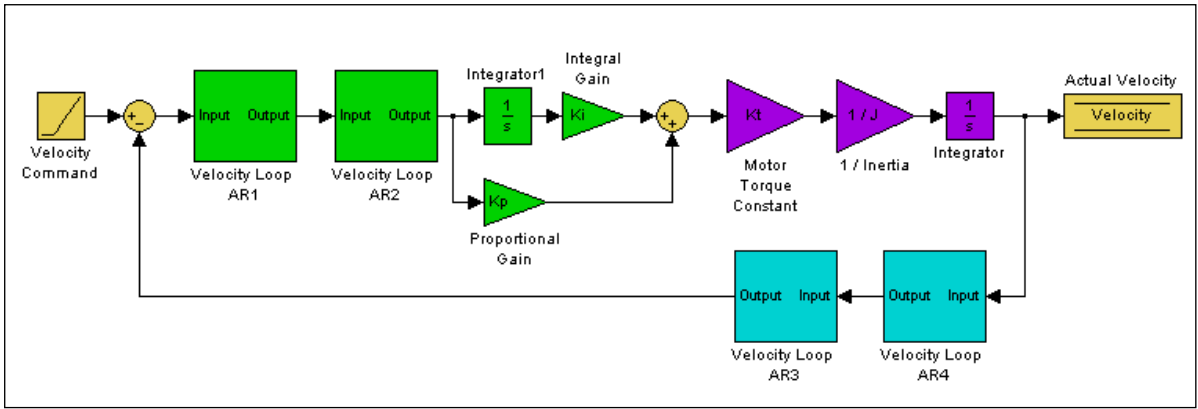
Reading and Understanding the Bode Plot

You can operate the PST without understanding how to read a Bode plot; however, understanding Bode plots will help you to use more advanced tuning techniques, which are covered more in depth in the Tuning Guide (→ p. 132) documentation.

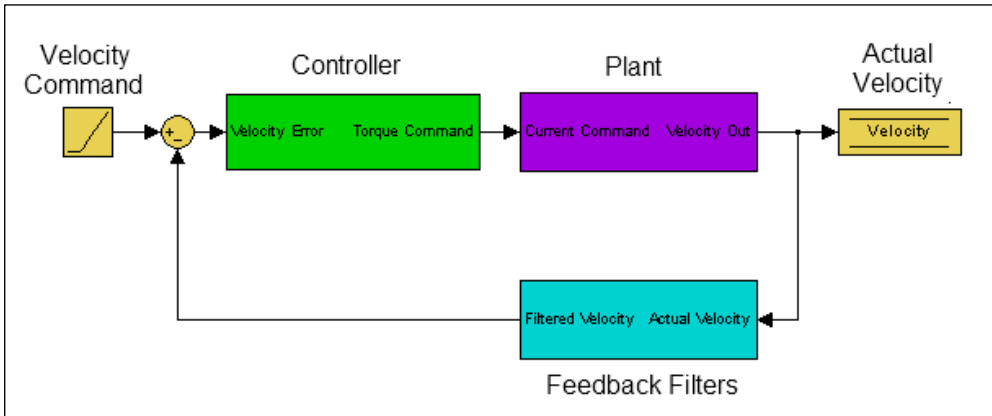
Four Bode plot traces are displayed by default:

1. Controller (green): This trace represents the frequency response of the tuning in the velocity loop and position loop, this trace also includes anti-resonance filter 1 and 2 (also referred to as [C]).
2. Closed loop (red): This trace shows the frequency response of $G/(1 + G * H)$ where $G = C * P$, and H is the frequency response of anti-resonance filters 3 and 4.
3. Open loop (purple): This trace shows the frequency response of $G * H$, where $G = C * P$, and H is the frequency response of anti-resonance filters 3 and 4.
4. Plant: This trace shows the frequency response of the mechanics of the drive and motor (also referred to as [P]).

The diagram of the velocity loop on the drive below explains the frequency response that each of these traces represents: Tuning Guide (→ p. 132)



These blocks can be grouped into Controller, Plant, and Feedback sections:



All of the green blocks have been grouped together to create the Controller [C]. The Controller is the portion of the control loop containing all velocity and position loop tuning, including the forward path filters.

All of the purple blocks have been combined to make the Plant [P]. The plant represents the mechanical and electrical properties of the motor, drive and any mechanical bodies attached to the motor.

The two feedback filters have been combined into one block. This value is never measured directly; however it contributes to both the Open Loop [G] and Closed Loop [T] frequency responses.

The definition of the Open Loop [G] frequency response is:

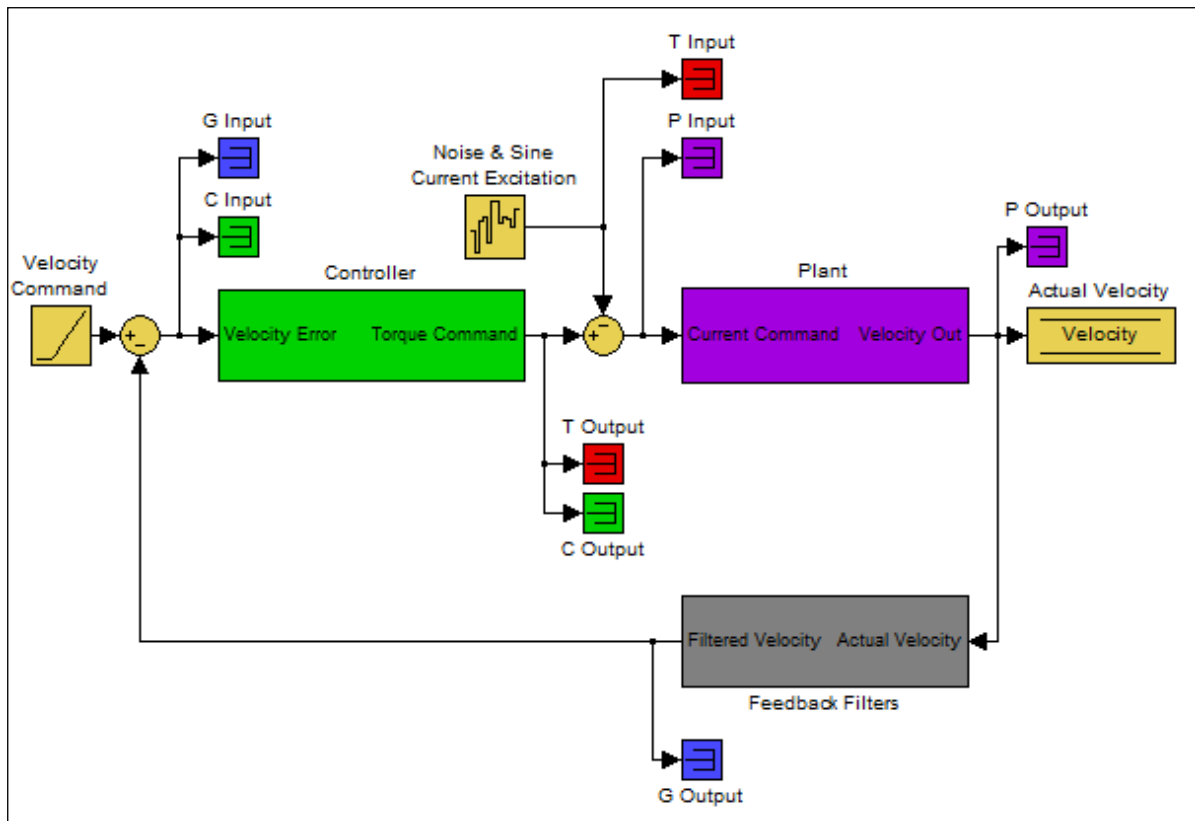
$$Open\ Loop = Controller \times Plant \times Feedback\ Filters$$

The definition of the Closed Loop [T] frequency response is:

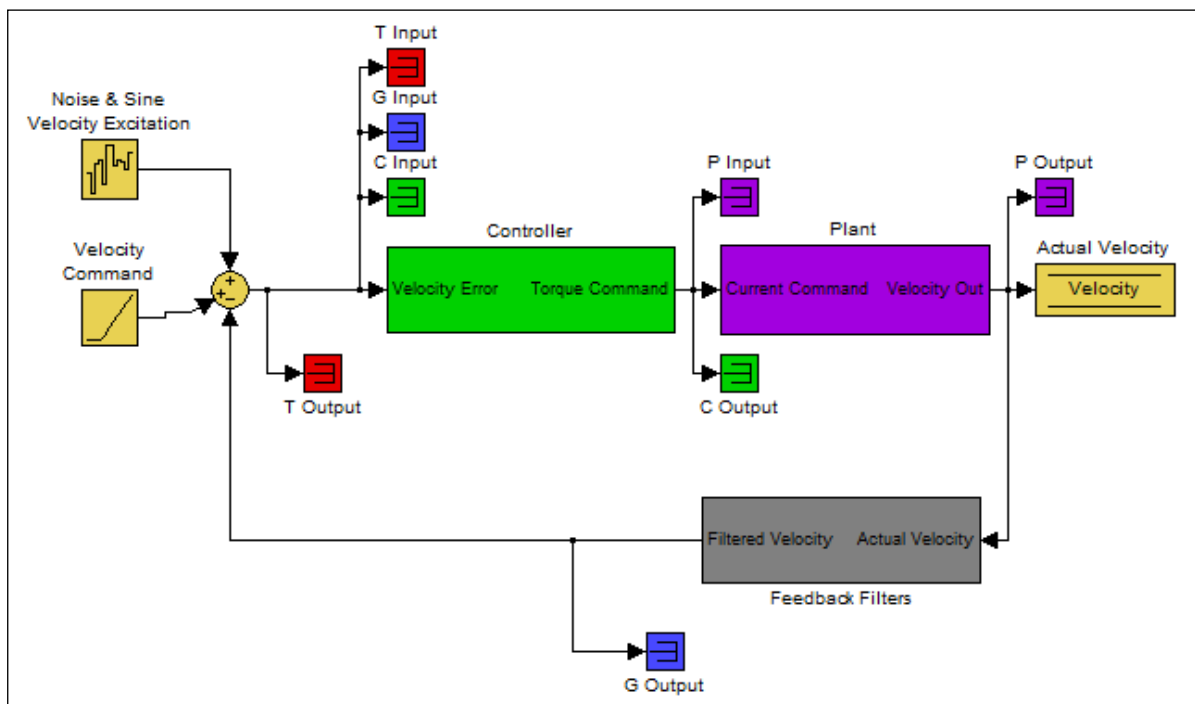
$$ClosedLoop = \frac{Controller \times Plant}{1 + Controller \times Plant \times FeedbackFilters}$$

Below is a diagram of measurement points (input and output) for each of these frequency responses. The input and output markers have been color coded with the color they appear in the PST:

Current Excitation:



Velocity Excitation:

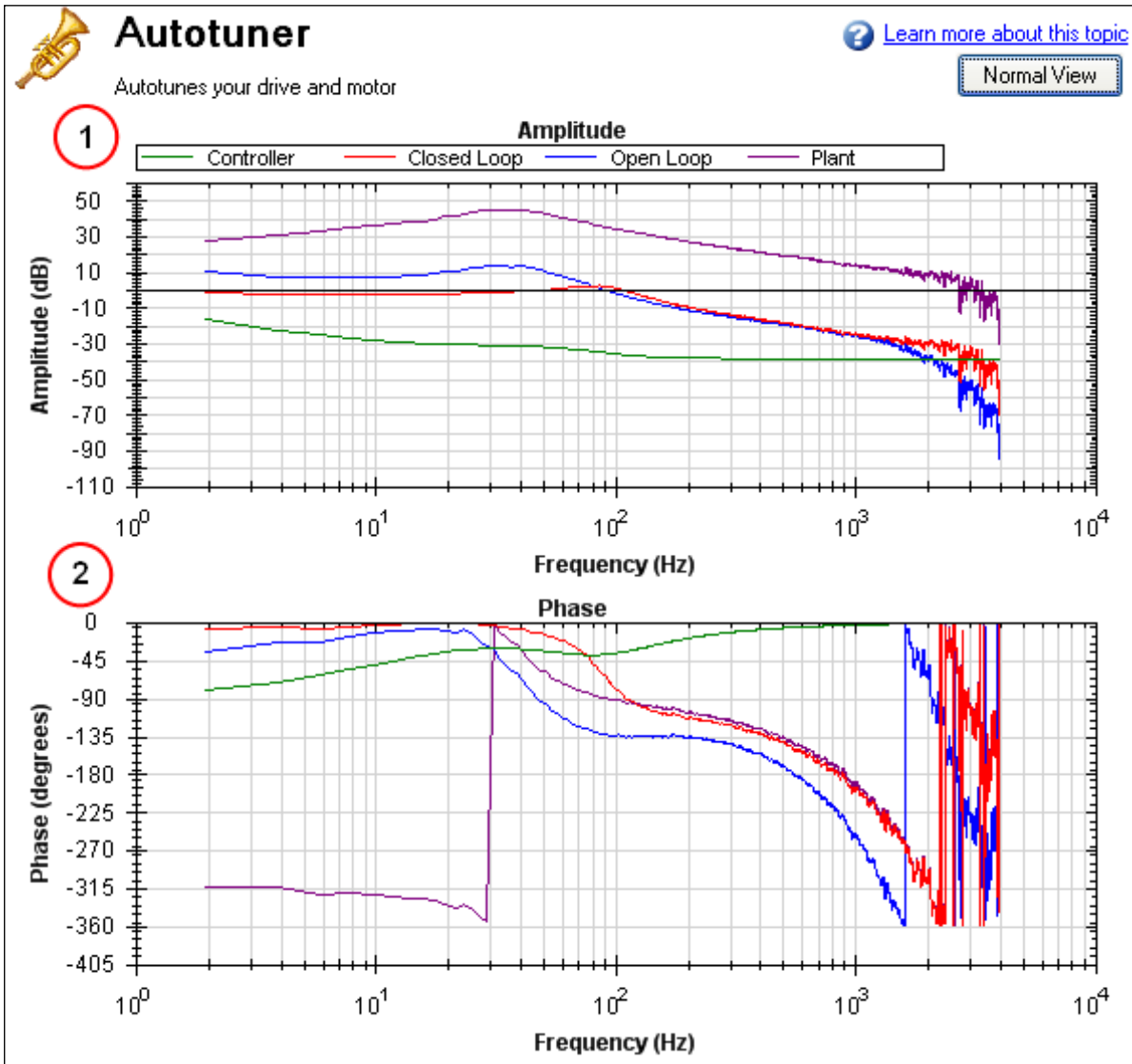


The resulting plots are the frequency response of output/input for each measurement.

For more information regarding these traces, please refer to the Tuning Guide (→ p. 132) documentation.

Below is a Bode plot of a motor with no load. The top plot is the magnitude plot (1); this plot shows the gain of the system with respect to frequency. This plot is often used to determine the bandwidth of the servo system.

The lower plot is the phase plot (2). This plot is used in conjunction with the magnitude plot to determine stability, and helps you to understand what kind of latencies exist in the servo system, or if latencies are induced by filters in the velocity loop.



Using the Performance Servo Tuner to Manually Tune Systems

Often, you must manually adjust a control loop in order to obtain optimal machine performance. You can use the Performance Servo Tuner (PST) interface to tune your control loop for best performance. A powerful feature of the manual tuning interface is the ability to simulate the frequency response before it is measured. This feature allows the user to take a base measurement, disable the motor, adjust tuning parameters, and simulate the frequency response of the motor without taking a new measurement. This process saves time and protects equipment from dangerous oscillations.

To begin the manual tuning process, put the Performance Servo Tuner into Bode Plot mode.

Performance Servo Tuner

Autotunes your drive and motor

[Learn more about this topic](#)

Full View

Tuning Options

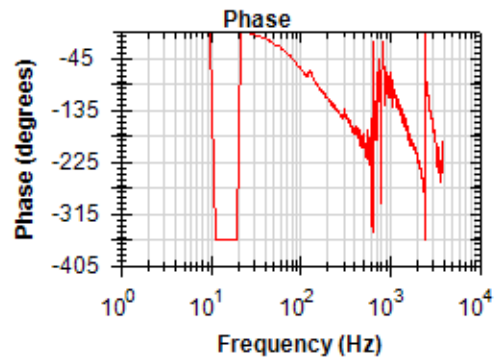
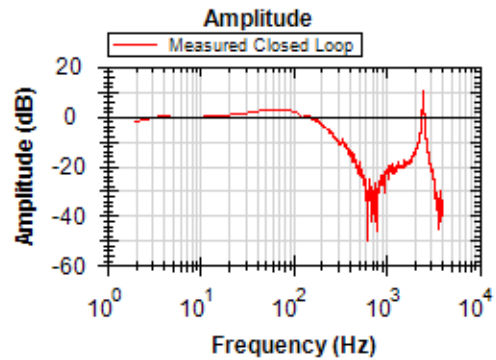
Excitation Level: Manual ▶ ▶▶ Start

Mode: Bode Plot ▶ **1**

Mouse Function: Zoom ▶

Progress: Complete ▶ Complete: ●

Less << Default Settings Import Save Email



Measurement | **Tuning** | Plots | Cursors | Note

Current Amplitude: 0.095 Measurement: Closed Loop ▶

Velocity Amplitude: 6.600 Injection Point: Current ▶

Smoothing %: 1.000 Excitation Type: PRB ▶

FFT Points: 4096 ▶ Velocity Max: 1000.00 (RPM)

Number Points: 60000 Excite Gap: 2

Several differences exist between PST and Bode Plot Interfaces:

- When the PST is put into Bode Plot mode, the **Autotuner** tab is removed from the advanced features, and replaced with a **Tuning** tab.
- The **Plots** tab unlocks simulated traces for closed loop, open loop, controller, and feedback filters.

Using the Tuning Simulation

To simulate tuning, there must be a valid Plant Plot in the PST (whether measured with a Bode Plot measurement or a full Autotune).

To selecting simulated plot traces, click on the **Plots** tab and check the following boxes:

Measurement | **Tuning** | **Plots** | Cursors | Note

Plot Plant

Plot Closed Loop Plot Simulated Closed Loop

Plot Open Loop Plot Simulated Open Loop

Plot Controller Plot Simulated Controller

Plot Feedback Filters Plot Simulated Feedback Filters

Plot Coherence

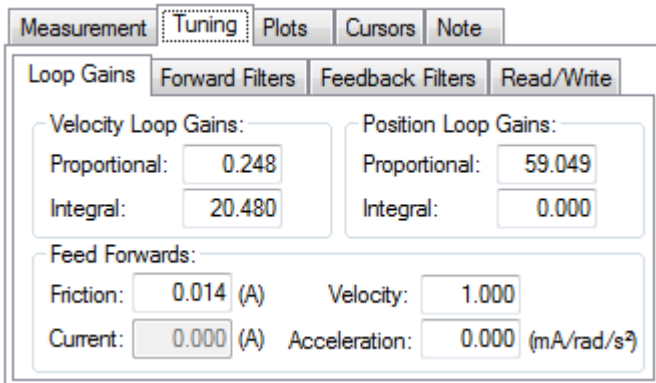
These selected boxes are the most common configuration for tuning; however, simulation will occur regardless of the checkboxes selected.

The boxes on the left plot the existing frequency response of the drive based on the tuning parameters that are loaded. The boxes marked "Simulated" (on the right) use the plant data from the measurement and the

tuning parameters in the PST to simulate the performance of those tuning parameters without loading them to the drive.

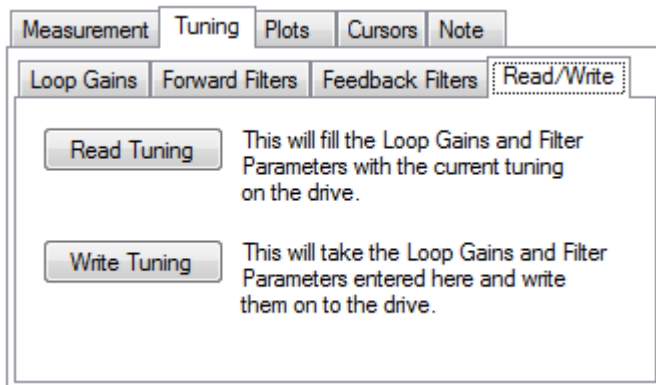
Using the Performance Servo Tuner Manual Tuning Interface

To use the PST manual tuning interface, click on the **Tuning** tab.



This tuning interface loads the tuning parameters on the drive each time a measurement is taken. Tuning parameters are split up into Loop Gains (Velocity Loop, Position Loop), Forward Path Biquad Filters, and Feedback Path Biquad Filters.

After modifying tuning gains, you must click on the **Read/Write** tab and click the **Write Tuning** button.

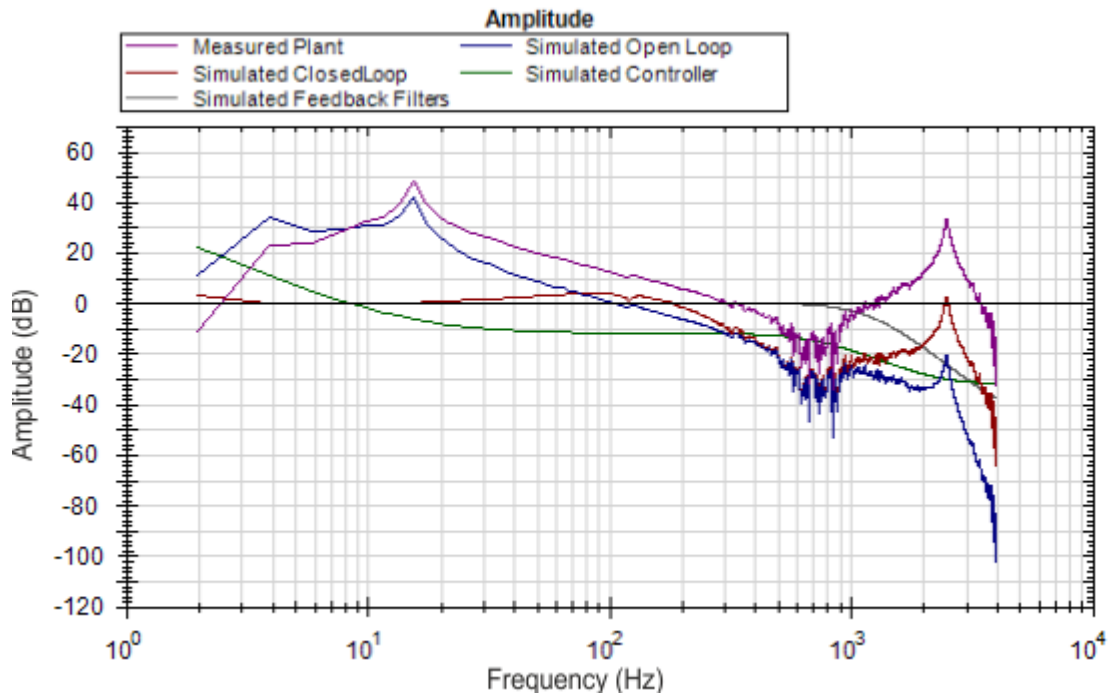


To restore the tuning on the drive to the PST interface, click the **Read Tuning** button.

Note: If tuning gains are modified and a Bode Measurement is made without clicking the **Write Tuning** button, the PST will overwrite the tuning gains in the interface with the tuning parameters on the drive.

Simulating Modified Loop Gains with the Performance Servo Tuner

Here is the frequency response of a test system after using the PST.

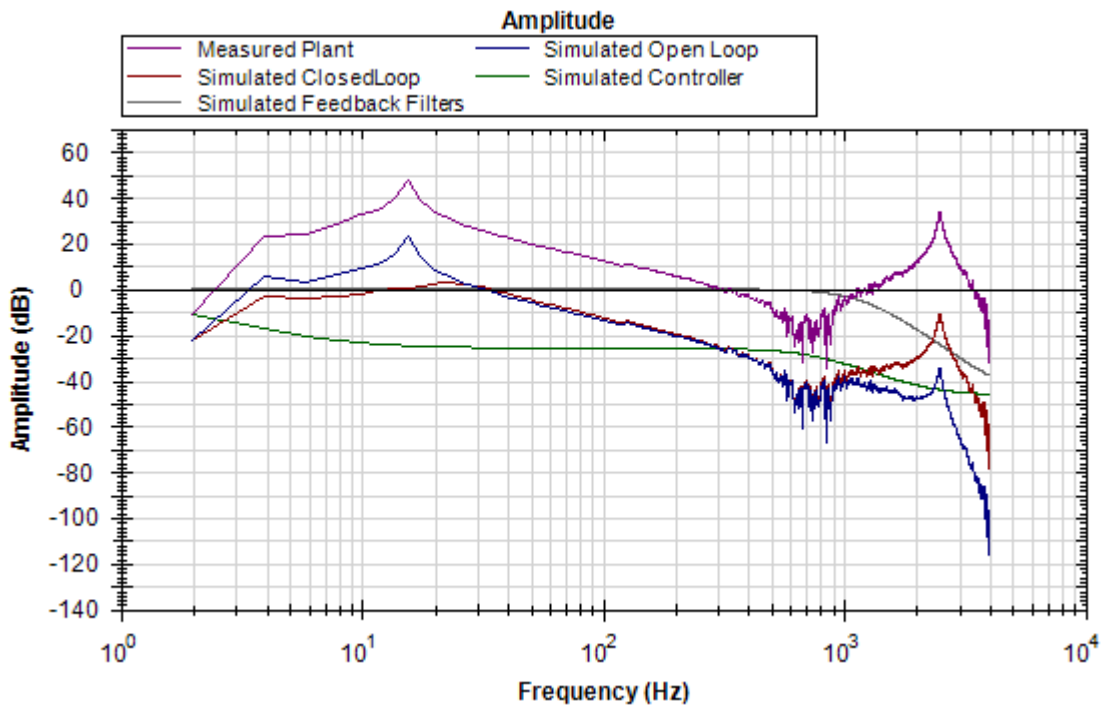


The Velocity Loop Proportional gain here is 0.248. If an application did not need to be tuned as stiff as this, then you could use the PST simulator to detune the motor to the desired bandwidth. A followup Bode Measurement can verify that the simulated response is correct.

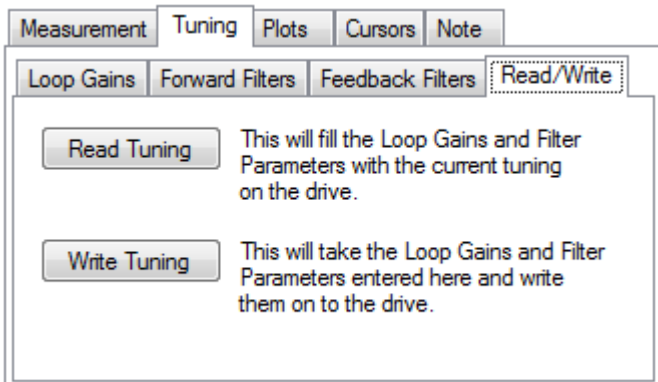
Use the boxes in the **Loop Gains** tab to change tuning gains until the desired frequency response is achieved.

| Measurement | | Tuning | | Plots | | Cursors | | Note | |
|----------------------|--|-----------------|---|----------------------|--|------------|--|------|--|
| Loop Gains | | Forward Filters | | Feedback Filters | | Read/Write | | | |
| Velocity Loop Gains: | | | | Position Loop Gains: | | | | | |
| Proportional: | <input type="text" value="0.050"/> | Proportional: | <input type="text" value="59.049"/> | | | | | | |
| Integral: | <input type="text" value="1.000"/> | Integral: | <input type="text" value="0.000"/> | | | | | | |
| Feed Forwards: | | | | | | | | | |
| Friction: | <input type="text" value="0.014"/> (A) | Velocity: | <input type="text" value="1.000"/> | | | | | | |
| Current: | <input type="text" value="0.000"/> (A) | Acceleration: | <input type="text" value="0.000"/> (mA/rad/s ²) | | | | | | |

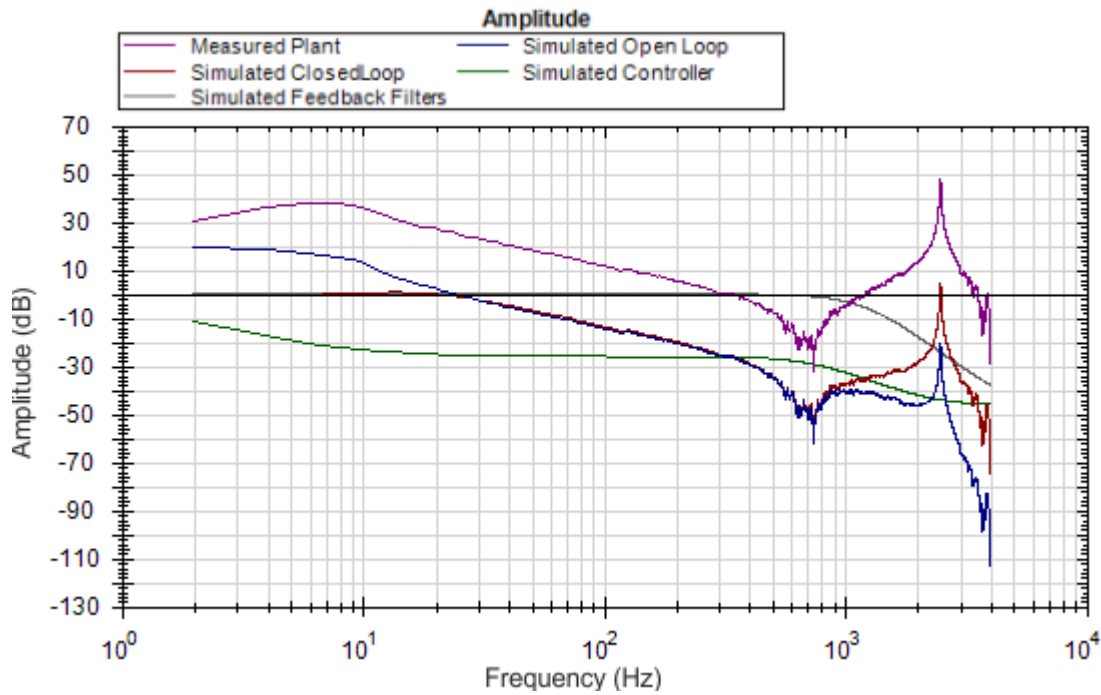
The de-tuning of velocity loop proportional and integral gains simulated that the bandwidth of the servo has been detuned from ~100 Hz to ~30 Hz.



Next, write the tuning parameters to the drive using the **Write Tuning** button on the **Read/Write** tab.



Now, complete a Bode Plot measurement to compare the simulated result with the new measured result.



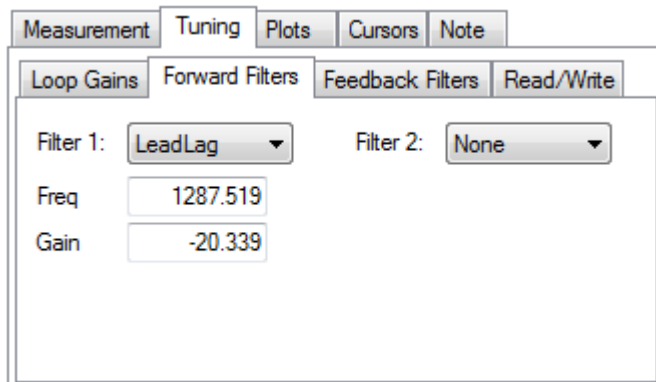
The new measured Bode Plot indicates we achieved slightly lower than 30 Hz bandwidth. The servo is stable, and tuning can be refined until desired performance is reached.

Simulating Filters with the Performance Servo Tuner

Resonances add many challenges to tuning a servo. Using the correct filter in an application can greatly improve system performance when resonances are present.

The Bode plot in this example shows a sharp, high-magnitude resonance at 2500 Hz. Because this is the only resonance, this is an indicator that a resonator (a tunable notch) filter may increase performance.

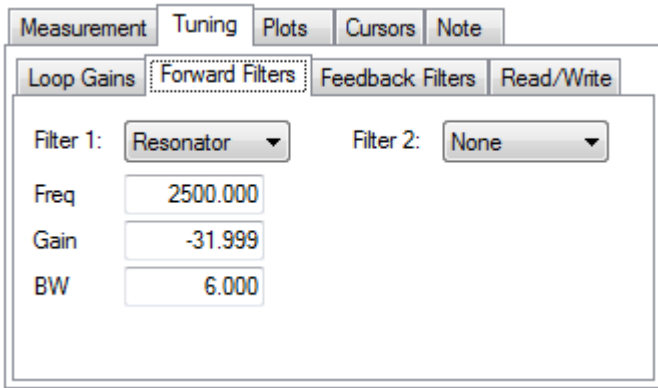
Click on the **Forward Filters** tab:



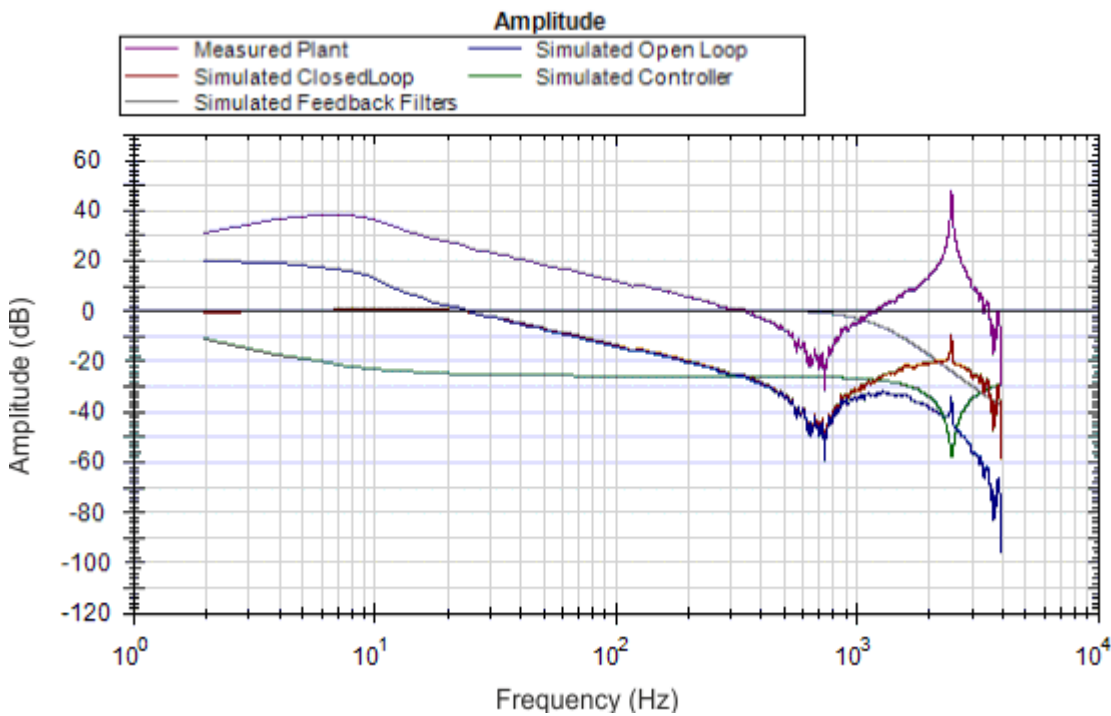
The results of the autotune are still on the drive, and provide adequate tuning. A lead lag filter is the default tuning filter, and is a good general case filter for most servo loops.

Because this test fixture has such a prominent single resonance, we can improve performance (and reduce noise) by placing a notch filter at this resonance.

By tuning a Resonator to best cancel the resonance in the plant, the resonance in the open loop, and therefore the closed loop can be minimized.



The resulting frequency response using the above resonator configuration is shown below:



Notice the attenuation of the resonance in the blue and red traces (open loop and closed loop, respectively).

Using Filters to Reduce Noise

To reduce noise, it is best to place filters in the feedback path. This placement attenuates the noise resulting from a noisy encoder being amplified by the current loop. This noise can be filtered by a forward path filter, however if a filter is placed in the forward path that introduces phase lag (like a lowpass), then your motion profile will exhibit that phase lag in the command signal. If the filter is placed in the feedback path, this lag will be avoided.

12.4 Tuning Guide

12.4.1 Overview

This section covers tuning the velocity and position loops in the AKD PDMM. Servo tuning is the process of setting the various drive coefficients that are needed for the drive to optimally control the servo motor for your application. There are different ways to tune, and several are covered here. We will give you guidance on what the different methods of tuning are and when to use them.

The AKD PDMM works in three major operation modes: torque, velocity, and position operation mode. No servo loop tuning is required for torque mode. Velocity loop and position loop tuning are covered below.

The AKD PDMM has an auto tuner that will provide the tuning that many applications will need. This section describes the tuning process and how to tune the AKD PDMM, specifically for cases where the user does not want to use the auto tuner.

Tuning in this section will focus on tuning in the time domain. This means that we will look at the velocity or position response vs. time as the criteria we use to decide how well tuned a control loop is tuned.

12.4.2 Determining Tuning Criteria

Choosing the proper specifications for a machine is a prerequisite for tuning. Unless you have a clear understanding of the type of performance needed to push the machine into production, the tuning process will cause more problems and headaches than it solves. Take time to layout ALL the requirements of the machine—nothing is too trivial to consider.

- Determine what the most important criteria are. The machine was likely designed and developed with a certain performance in mind. Include ALL performance criteria in the specification. Do not concern yourself with whether or not the criteria sound scientific. (i.e. If the motion needs to visibly look smooth, put it in the specification. If it can't have any noise, put it in the specification.) At the end of the development phase, the machine's performance should match the performance previously set in the specification. This will ensure that the machine meets its performance goals and that it is ready for production.
- Test the machine with realistic motion. Do not simply tune the machine to make short linear motion, when it will make long, s-curve motions in the real world. Unless you test the machine with realistic motion, there is no way to determine if it is ready for production.
- Determine some specific, quantitative criteria for identifying unacceptable motion. It's better to be able to tell when a motion is unacceptable than to try and figure out the exact point where acceptable motion becomes unacceptable. Here are some examples of motion criteria:
 1. +/- x position error counts during the entire motion.
 2. Settling within +/- x position error counts, within y milliseconds.
 3. Velocity tolerance of x% measured over y samples.
- It is important to focus on the things that will get the machine into production with reliable performance, based on a fundamental understanding of the system.

After you have constructed a detailed servo performance specification, you are now ready to start tuning your system.

12.4.3 Before You Tune

In the worst case, if something goes wrong during tuning, the servo can run away violently. You need to make sure that the system is capable of safely dealing with a servo run away. The drive has several features that can make a servo run away safer:

- Make sure that the limit switches turn the drive off when tripped. If a complete run away occurs, the motor can move to a limit switch very quickly.
- Make sure the max motor speed is set accurately. If a complete run away occurs, the motor can reach max speed quickly and the drive will then disable.

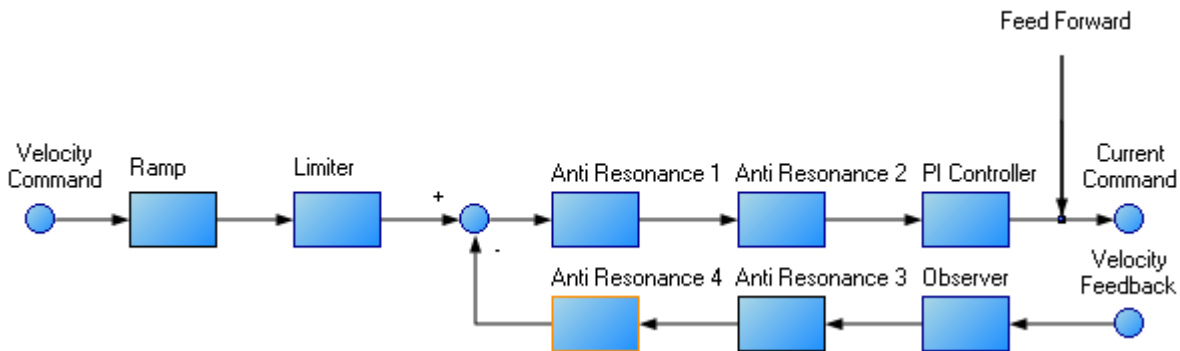
12.4.4 Closed Loop Tuning Methods

The closed loop control loop is responsible for the desired position and / or velocity (trajectory) of the motor and commanding the appropriate current to the motor to achieve that trajectory. The challenge in closed loop control loops is to make a system that not only follows the desired trajectory, but also is stable in all conditions and resist external forces, and do all of this at the same time.

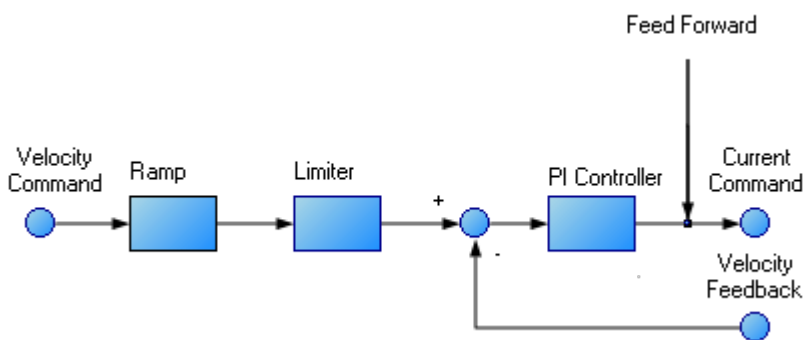
When in velocity operation mode, only the velocity loop is tuned. When in position operation mode, both the velocity and position loops must be tuned.

12.4.4.1 Tuning the Velocity Loop

The velocity loop on the AKD PDMM consists of a PI (proportional, integral) in series with two anti-resonance filters (ARF) in the forward path and two anti-resonance filters in series in the feedback path.



To perform basic tuning of the velocity loop, you can use just the PI block and set ARF1 and ARF2 to unity (no effect) and set the observer to 0 (no effect). Using just the PI block simplifies the process of tuning the velocity loop. To start tuning you can adjust the PI Controller block first. A simplified velocity loop without anti-resonant filters and observer is shown below. This is how you can think of the loop before the anti-resonant filters and observer is used.



Procedure for simple velocity loop tuning:

1. Set DRV.OPMODE to velocity or position, as appropriate for your application. If DRV.OPMODE is set to position, set VL.KVFF to 1.0.
2. Set VL.KP to 0.
3. Set VL.KI to 0.
4. Set service motion to make a motion that is similar to the move speeds that will be used in the real application. Do not set the service motion to a speed higher than $\frac{1}{2}$ of the maximum motor speed, to allow for safe overshoot during tuning. Set acceleration to an appropriate value for your application. Set service motion to reversing. Set time1 and time2 equal to 3 times the expected settling time for the system. 1.0 second is a reasonable value for time1 and time2, if you don't know the expected settling time.
5. Enable the drive and start the service motion. You should see no motion, as there are no velocity loop tuning gains at this point.
6. When adjusting VL.KP and VL.KI, below record VL.FB and VL.CMD. These are the traces that are used to determine the performance of the velocity loop.

| Channels | | | | | | |
|-----------------------|-----------------------------|--------|--------------------------|----------|--------------------------|----------------|
| Time-base and Trigger | | | | | | |
| Service Motion | | | | | | |
| Servo Gains | | | | | | |
| Observer | | | | | | |
| All Gains | | | | | | |
| AR F | | | | | | |
| Id | Source | Color | Hide | Y Axis | Filter | Filter Freq... |
| 1 | Current feedback (IL.FB) | Red | <input type="checkbox"/> | Current | <input type="checkbox"/> | 400 |
| 2 | Velocity command (VL.CM...) | Green | <input type="checkbox"/> | Velocity | <input type="checkbox"/> | 400 |
| 3 | Velocity feedback (VL.FB) | Blue | <input type="checkbox"/> | Velocity | <input type="checkbox"/> | 400 |
| 4 | None | Purple | <input type="checkbox"/> | Default | <input type="checkbox"/> | 400 |
| 5 | None | Orange | <input type="checkbox"/> | Default | <input type="checkbox"/> | 400 |
| 6 | None | Yellow | <input type="checkbox"/> | Default | <input type="checkbox"/> | 400 |

Idle

Stop Motion

Enable Drive

Start Recording

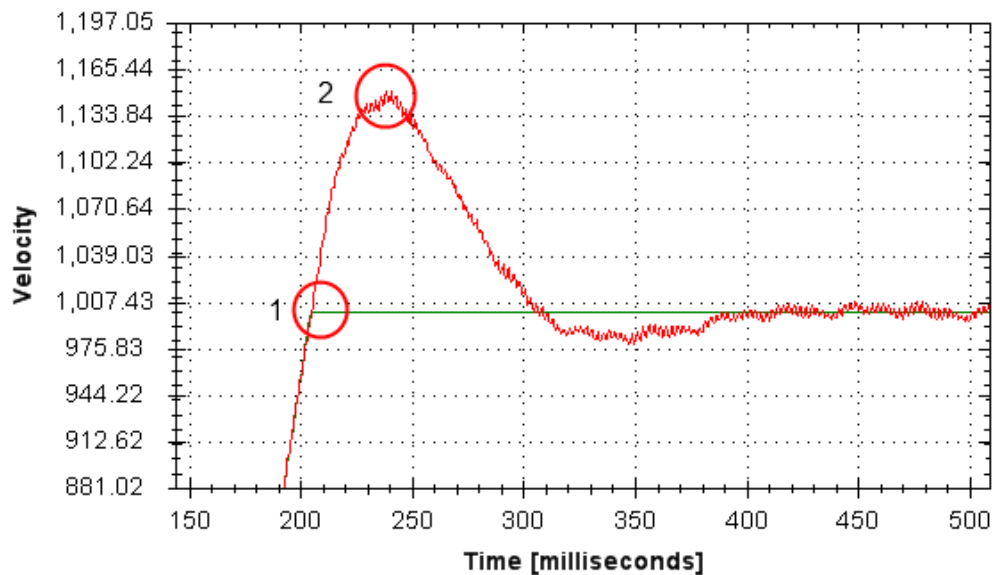
Refresh

7. Adjust VL.KP. Keep increasing VL.KP by a factor of 2 until you either:

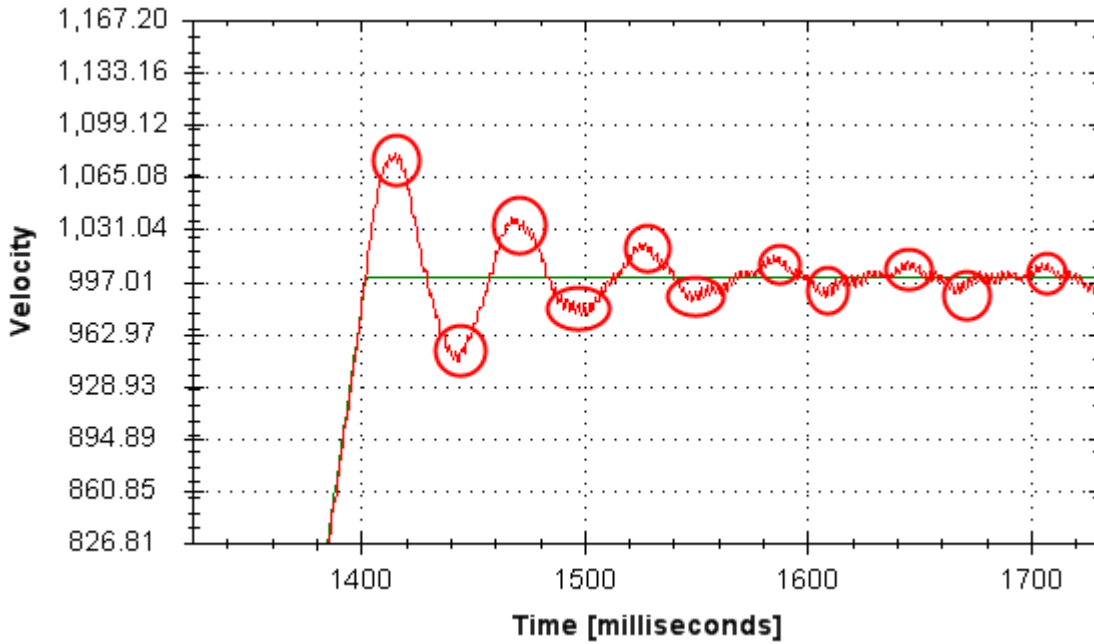
- Hear an objectionable noise from the system (buzzing, humming, etc) or
- See velocity overshoot. No velocity overshoot should be present when using only VL.KP.
- When you reach one of the limits above, decrease VL.KP to the value where there were no objectionable noises or overshoot.

8. Adjust VL.KI. Increase VL.KI by a factor of 1.5 until you either:

- Hear or see objectionable noise or shuddering from the system
- See > 15% overshoot
- Here is an example of 15% overshoot. This is zoomed in view of a service motion commanded to 1000 RPM (location 1), where the overshoot peaks at 1150 RPM (location 2).



- Here is an example of 11 overshoots. Each overshoot is shown by a red circle.



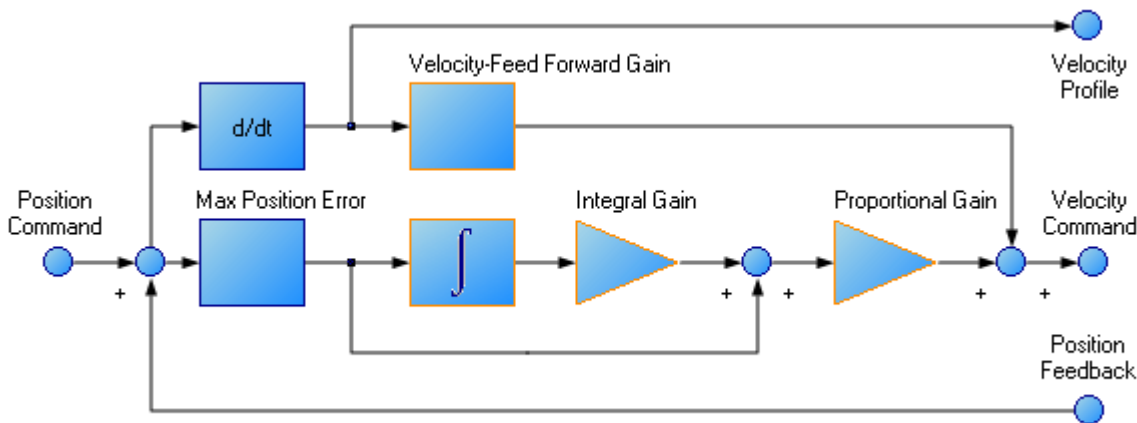
- When you reach one of the limits above, decrease VL.KI to the value where there were no objectionable noises or overshoot.

9. Stop the service motion

12.4.4.2 Tuning the Position Loop

The position loop is a second loop that builds upon a correctly tuned velocity loop to provide accurate control over position. The position loop is a simple element that consists of a PI loop. It is simplest to tune the P and I terms in the velocity loop and use only the P term in the position loop.

At most, use only three non-zero P and I terms from both the velocity loop and the position loop. One combination would be VL.KP, VL.KI, and PL.KP. Another valid combination would be VL.KP, PL.KP, and PL.KI. The VL.KP, VL.KI, and PL.KP combination is shown here.



Procedure for tuning position loop:

- Set VL.KVFF to 1
- Increase PL.KP until either:
 - You see 25% overshoot, or
 - You see > 3 overshoots, or

- You hear objectionable noises from the system.
- When you reach one of the limits above, decrease PL.KP to the value where there were no objectionable noises or overshoot.

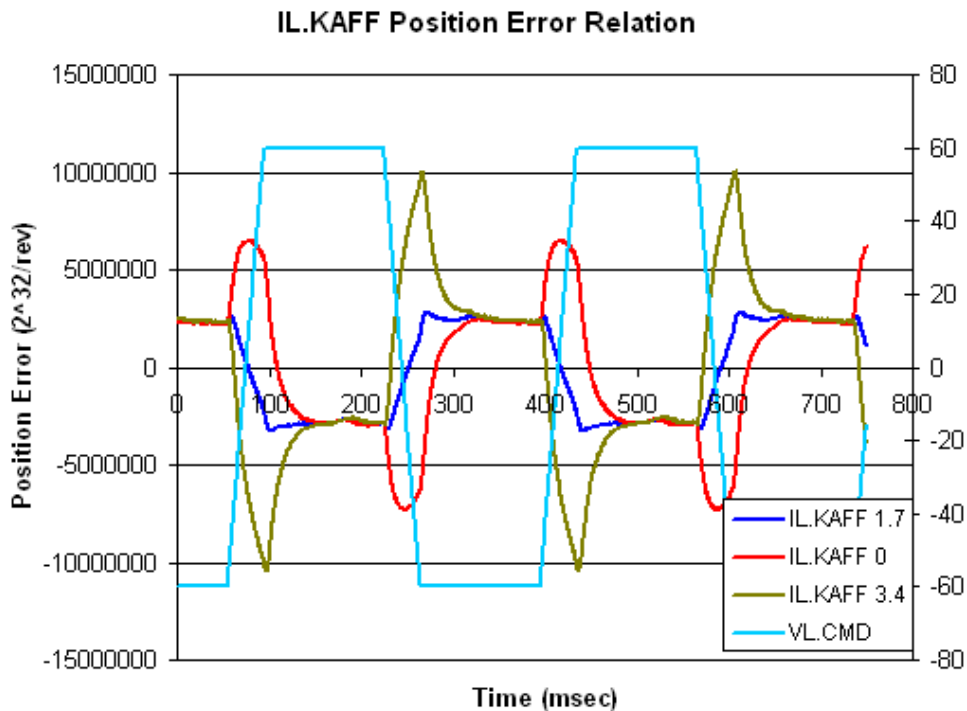
12.4.5 Torque Feedforward Tuning Methods

The torque based feedforward terms on the AKD PDMM effectively model the physics of your motor and allow the drive to command the appropriate current, even before the encoder has time to send data back to the drive. Torque based feedforward terms allow you to lower following error with virtually no stability penalty.

12.4.5.1 Shape Based Feedforward Tuning

To adjust IL.KAFF:

- Tune the VL.KP and VL.KI as shown above in the velocity loop tuning section. Set DRV.OPMODE to velocity (or set PL.KP and PL.KI to 0 and vl.kvff to 1).
- Set up a short, repeating service motion with accelerations that are representative of the moves you will use in your application (exact values for acceleration are not critical).
- Turn up IL.KAFF until the position error (PL.ERR) is proportional to the inverted velocity command. The adjustment of IL.KAFF will focus on removing bumps on acceleration and deceleration. The picture below has an ideal value of IL.KAFF of 1.7.



12.4.6 Using Anti-Resonance Filters

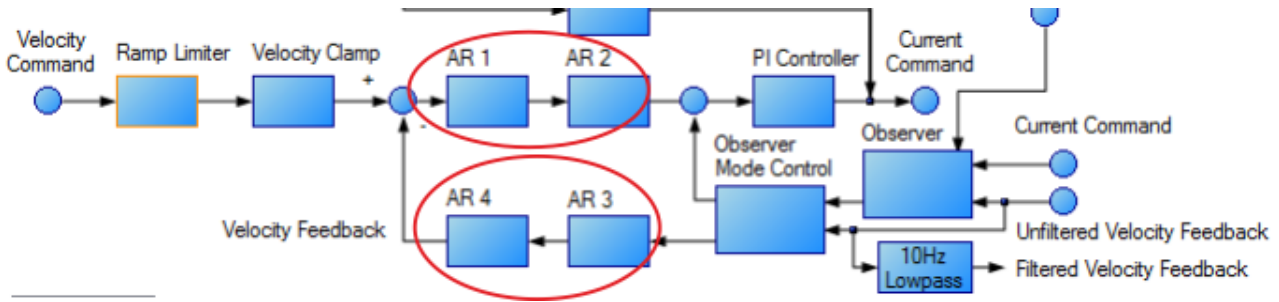
The AKD PDMM has four anti-resonance filters. Two filters are in the forward path and two are in the feedback path.

Similarities

- Both types are typically used to enhance stability and performance of the system.

Differences

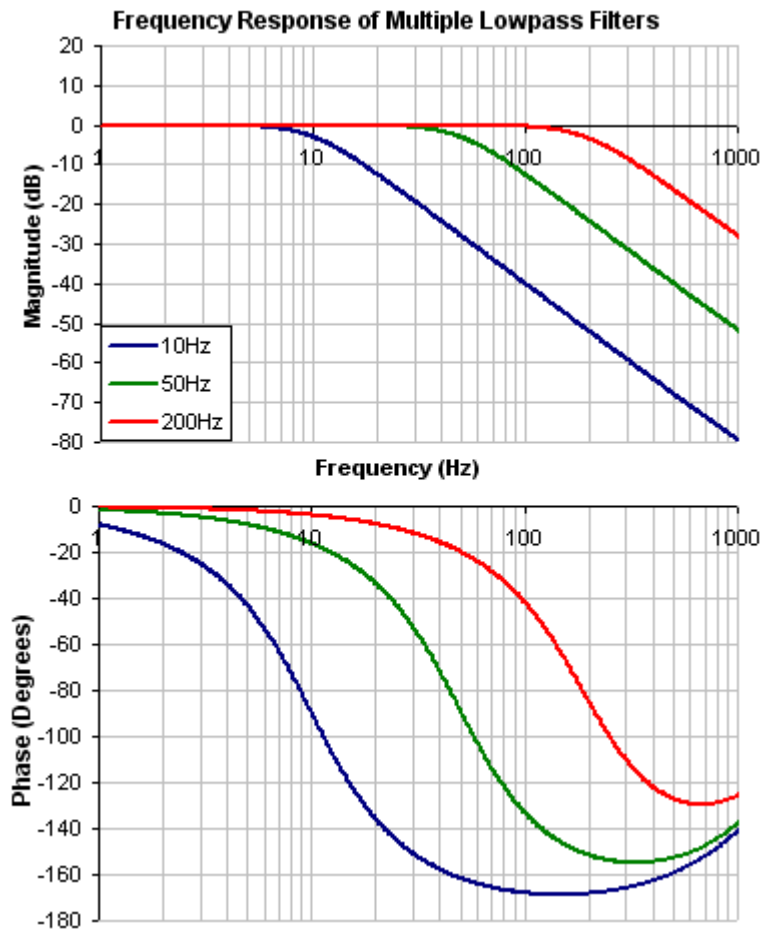
- Forward path filters result in higher phase lag in closed loop system response.
- Forward path filters limit spectrum from reaching the motor / feedback path filters only filter the feedback after it has been to the motor.



12.4.6.1 Types of Anti-Resonance Filters

Low Pass

A low pass filter allows signals through below a corner frequency and attenuates the signals above the same corner frequency. The behavior at the corner frequency can be specified with the low-pass Q.



To specify a lowpass filter, you must specify the frequency and Q for both the zero and pole on anti-resonance filter 1. To do this, see the following example using the terminal commands that sets:

- Filter Type = Biquad
- Zero frequency = 700 Hz (This is the Lowpass cutoff frequency)
- Zero Q = 0.707

- Pole frequency = 5000 Hz
- Pole Q = 0.707

VL.ARTYPE1 0

VL.ARZF1 700

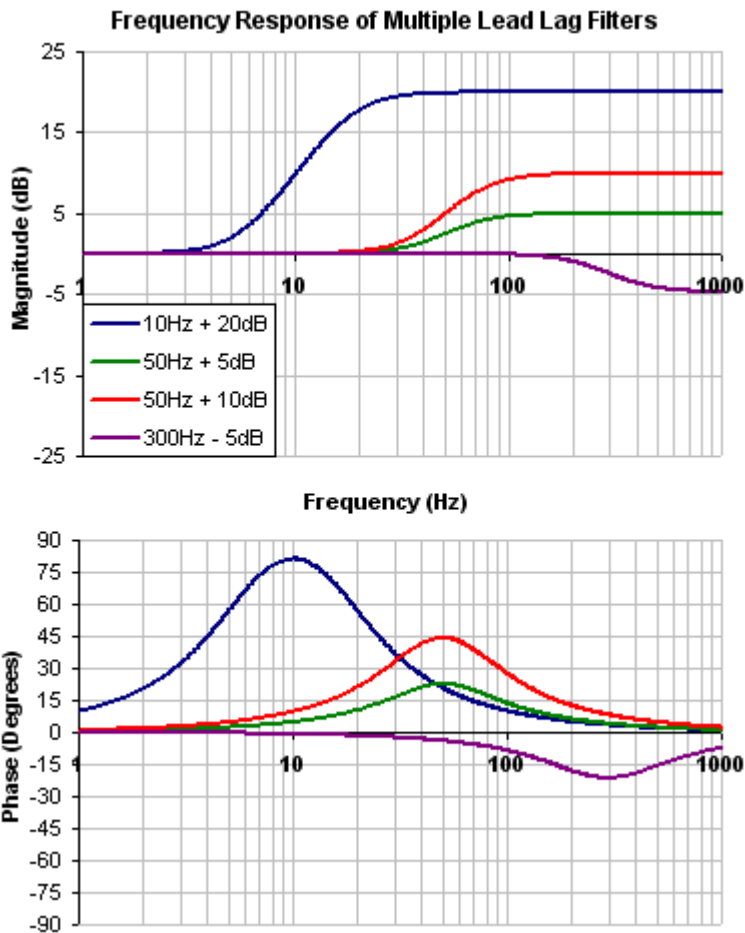
VL.ARZQ1 0.707

VL.ARPF1 5000

VL.ARPQ1 0.707

Lead Lag

A lead lag filter is a filter that has 0 dB gain at low frequencies and a gain that you specify at high frequencies. You also specify the frequency that the gain at which the transition occurs.



To specify a Lead Lag filter, you must specify the Center Frequency and high frequency Gain (dB). To do this, see the following example by clicking on the Velocity Loop:

Click on Velocity Loop tab (1), then select the AR1 Tab (2), using the Filter Type drop-down, select Lead Lag (3), lastly, enter the desired Center Frequency and Gain of the Lead Lag filter (4).

Velocity Loop
The parameters for controlling the velocity of the motor.

The diagram shows the following components and flow:

- Velocity Profile** and **Velocity Command** inputs.
- Acceleration Feed Forward** block (d/dt).
- Friction Compensation** block (SIGN).
- Viscous Damping Compensation** block.
- Ramp Limiter** and **Velocity Clamp** blocks.
- AR 1** and **AR 2** blocks (AR Filter).
- PI Controller** block.
- Observer Mode Control** block.
- Observer** block.
- 10Hz Lowpass** filter.
- Observer Bandwidth** and **Current Command** outputs.
- Unfiltered Velocity F** and **Filtered Velocity Fee** outputs.

AR Filter Configuration:

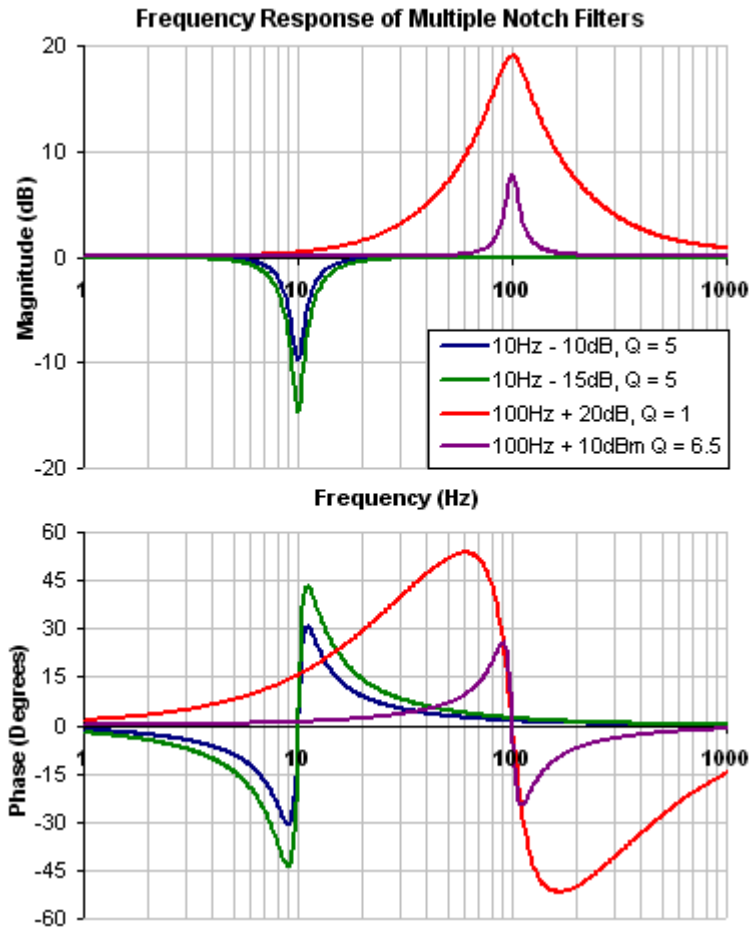
| AR | Type |
|------|------------|
| AR 1 | Lead Lag |
| AR 2 | Unity Gain |
| AR 3 | Low Pass |
| AR 4 | Unity Gain |

Filter Parameters:

- Filter Type: 3 - Lead Lag
- Center Frequency: 500,000 Hz
- Gain (dB): 0,000
- Numerator: Frequency: 500,000 Hz, Q: 0,500
- Denominator: Frequency: 500,000 Hz, Q: 0,500

Notch

A notch filter changes gain at a specific frequency. You specify the frequency at which the gain change occurs (Frequency (Hz)), how wide of a frequency range the cut occurs (Q), and how much the gain changes (Notch Depth (dB)).



To specify a notch filter, you must specify the Frequency (Hz), Depth (dB) and Width (Q) of the notch. To do this, see the following example by clicking on the Velocity Loop:

Click on Velocity Loop (1), then select the AR1 Tab (2), using the Filter Type drop-down, select Notch (3), lastly, enter the desired Frequency, Depth and Q of the Notch filter (4).

Velocity Loop
The parameters for controlling the velocity of the motor.

| AR | Type |
|------|------------|
| AR 1 | Notch |
| AR 2 | Unity Gain |
| AR 3 | Low Pass |
| AR 4 | Unity Gain |

Filter Type: 2 - Notch

Edit Parameters

Frequency (Hz): 500.000

Depth (dB): 0.000

Q: 0.600

Biquad

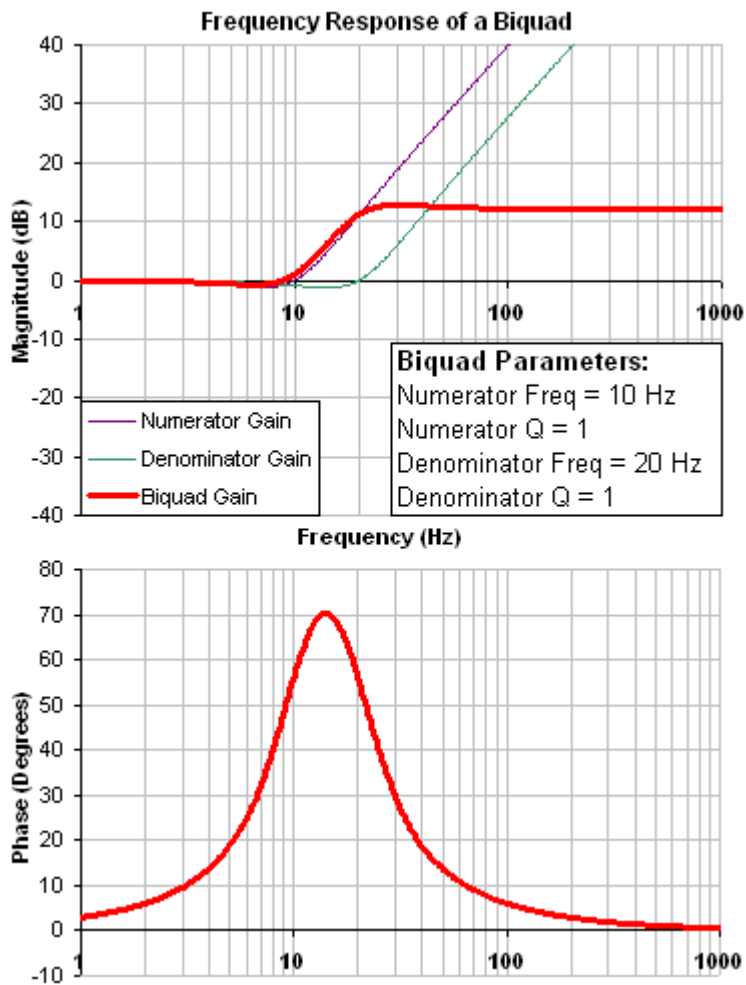
A biquad is a flexible filter that can be thought up as being made up of two simpler filters; a zero (numerator) and a pole (denominator). In fact, the pre-defined filters mentioned above are really just special cases of the biquad.

Both the zero (numerator) and the pole (denominator) have a flat frequency response at low frequencies and a rising frequency response at high frequencies. The transition frequency and damping must be specified for both the numerator and denominator.

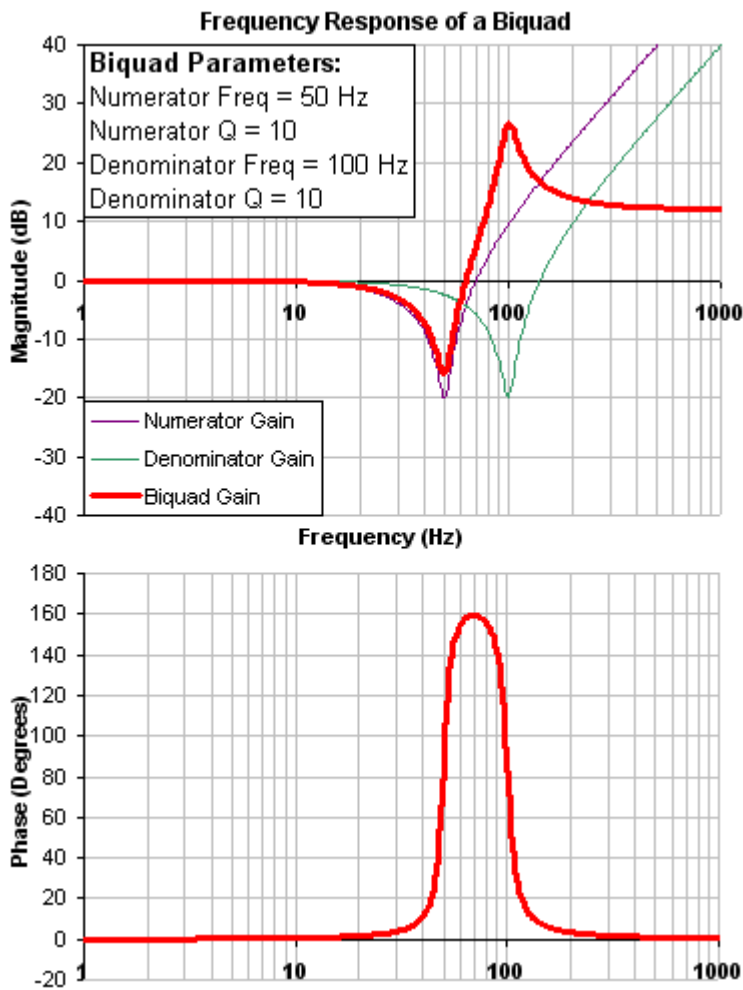
Analyzing the numerator and denominator, the frequency response calculation is simple:

If the numerator and denominator are plotted in dB, the biquad response is numerator – denominator. Understanding how the numerator and denominator work is crucial in understanding how a biquad frequency response is created.

Below is an example of a biquad filter similar to a Lead Lag filter type. To help understand how to determine the frequency response of the biquad, the numerator and denominator response have been plotted. If the denominator is subtracted from the numerator, the biquad response is the result.

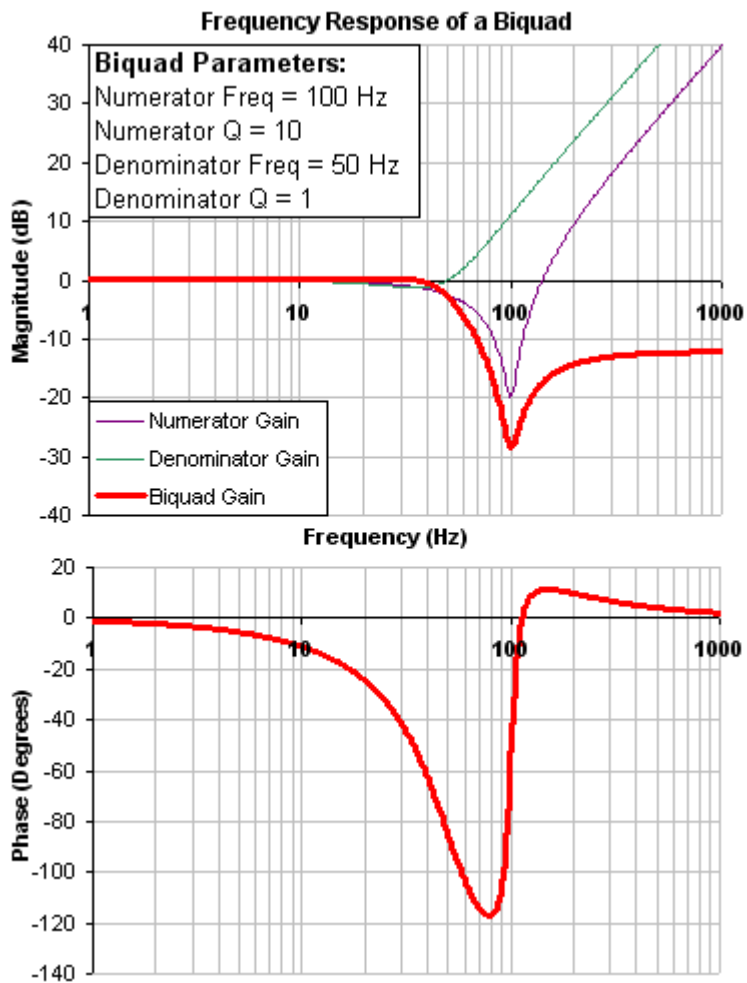


The biquad filter is very flexible, which allows custom filters to be designed. Below is an example of a resonance filter using a biquad. Notice how the high Q values affect the numerator and denominator. This gives a biquad frequency response similar to a mechanical resonance.



The previous two examples used a numerator frequency lower than the denominator frequency, yielding a positive gain in high frequencies. If the denominator frequency is lower than the numerator frequency, then high frequencies will have a negative gain.

Below is an example where the numerator frequency is higher than the denominator. Notice the high frequencies have a negative gain.



To specify a biquad filter, you must specify the frequency and Q for both the zero and the pole on anti-resonance filter 3. To do this, see the following example using the terminal commands that sets:

- Filter Type = Biquad
- Zero frequency = 100 Hz
- Zero Q = 0.7
- Pole frequency = 1000 Hz
- Pole Q = 0.8

```
VL.ARTYPE3 0
```

```
VL.ARZF3 100
```

```
VL.ARZQ3 0.7
```

```
VL.ARPF3 1000
```

```
VL.ARPQ3 0.8
```

12.4.6.2 Biquad Calculations

In the s-domain, the linear biquad response is calculated:

$$\text{Biquad Frequency Response} = \frac{s^2 + \frac{\omega_N}{Q_N} s + \omega_N^2}{s^2 + \frac{\omega_D}{Q_D} s + \omega_D^2}$$

To convert from idealized s-domain behavior to a more realistic z-domain behavior, we convert using a pole / zero transform. To calculate the frequency response for an individual frequency:

$$t = 62.5 \mu \text{ sec}$$

$$N_{Rad} = 1 - (2Q_N)^{-2}$$

$$N_{2Unscaled} = 1$$

$$\text{if}(N_{Rad} > 0): N_{1Unscaled} = -2e^{-2\omega_N \zeta_N t} \cos(\omega_N t \sqrt{1 - \zeta^2})$$

$$\text{if}(N_{Rad} \leq 0): N_{1Unscaled} = -2e^{-2\omega_N \zeta_N t} \cosh(\omega_N t \sqrt{1 - \zeta^2})$$

$$N_{0Unscaled} = e^{-2\omega_N \zeta_N t}$$

$$D_{Rad} = 1 - (2Q_D)^{-2}$$

$$D_2 = 1$$

$$\text{if}(D_{Rad} > 0): D_1 = -2e^{-2\omega_D \zeta_D t} \cos(\omega_D t \sqrt{1 - \zeta^2})$$

$$\text{if}(D_{Rad} \leq 0): D_1 = -2e^{-2\omega_D \zeta_D t} \cosh(\omega_D t \sqrt{1 - \zeta^2})$$

$$D_0 = e^{-2\omega_D \zeta_D t}$$

$$N_{Scale} = \frac{N_{0Unscaled} + N_{1Unscaled} + N_{2Unscaled}}{D_0 + D_1 + D_2}$$

$$N_2 = N_{2Unscaled} / N_{Scale}$$

$$N_1 = N_{1Unscaled} / N_{Scale}$$

$$N_0 = N_{0Unscaled} / N_{Scale}$$

$$\angle_z = \omega t = 2\pi * freq * t$$

$$Num_{Re} = N_2 \cos(2\angle_z) + N_1 \cos(\angle_z) + N_0$$

$$Num_{Im} = N_2 \sin(2\angle_z) + N_1 \sin(\angle_z)$$

$$Den_{Re} = D_2 \cos(2\angle_z) + D_1 \cos(\angle_z) + D_0$$

$$Den_{Im} = D_2 \sin(2\angle_z) + D_1 \sin(\angle_z)$$

$$Gain_{dB} = 20 \log_{10} \left(\frac{\sqrt{Num_{Re}^2 + Num_{Im}^2}}{\sqrt{Den_{Re}^2 + Den_{Im}^2}} \right)$$

$$Phase_{deg} = \frac{180}{\pi} \left(\tan^{-1} \left(\frac{Num_{Re}}{Num_{Im}} \right) - \tan^{-1} \left(\frac{Den_{Re}}{Den_{Im}} \right) \right)$$

12.4.6.3 Common Uses Of Anti-Resonance Filters

Low pass filters in the feedback path. This is a common way to deal with noisy feedback sensors. When used in combination with noisy feedback sensors, significant reduction in audible noise can result.

Lead / lag filters in the forward path. This is a common way to achieve phase lead for control loops without exciting high frequency resonances.

Low pass filters in the forward path. This is a common way to limit high frequency energy from reaching a system that can not productively use energy at these high frequencies. This is also used to lower the effect of system resonances over a wide range of frequencies.

Notch filters are used to cancel system resonances. Notch filters are designed to be the opposite in amplitude of system resonances. Notch filters are applied to very specific frequencies, and therefore you must know your system resonance frequencies accurately to use them effectively.

12.5 Cogging Compensation

Cogging compensation can be used to counteract cogging effects present in the motor by supplying an additional feedforward current. The result is a reduction of velocity ripple, mainly at lower velocities.

This is especially useful for applications with a high demand for synchronism.

The AKD PDMM uses a cogging compensation table that stores the feedforward values to apply depending on the motors position. The table consists of 8192 entries that are evenly distributed over one revolution (rotary axis) or a range that can be specified by the user (linear axis).

The drive can also automatically detect cogging effects and generate a fitting compensation table.

12.5.1 Controlling Cogging Compensation

12.5.1.1 Enabling

Cogging compensation can be enabled and disabled in the Compensation tab of WorkBench's Current Loop view (see below) or using the keyword COGCOMP.EN (→ p. 321). Enabling cogging compensation or booting the drive with Cogging compensation saved as enabled and using a non-absolute feedback while not being homed will display a warning and not apply compensation values. To clear the warning, either execute a homing procedure or deactivate Cogging compensation.

⚠ WARNING

Since cogging compensation values directly translate to current supplied to the drive you should always be sure that no unusually high compensation values are stored in the table.

Current Loop
Parameters for controlling the torque/force of the motor. Usually no cl

Velocity Command → Coulomb Friction Comp.

Current Command + Current Offset → Summing Junction → Cogging Comp. → Limiter → Output

| Compensation | PI Controller | Limiter | Source | Status |
|----------------------|--|---------|--------|--------|
| Offset | | | | |
| Current Offset: | 0.000 A | | | |
| Friction | | | | |
| Friction Current: | 0.014 A | | | |
| Viscous FF Gain: | 0.000 mAms/(rad/s) | | | |
| Cogging Compensation | | | | |
| Mode: | 0 - Disabled Goto Cogging Compensation | | | |

12.5.1.2 Cogging Compensation View

The cogging compensation view can be accessed by expanding the Current Loop item in WorkBench's Device Topology list and selecting Cogging Compensation or by following the Goto Cogging Compensation Link in the Compensation tab of WorkBench's Current Loop view.



Cogging Compensation

Advanced user cogging compensation table.

Compensation Table

| Table Index | Compensation (mA) |
|-------------|-------------------|
| 0 | 0 |
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | 0 |
| 7 | 0 |
| 8 | 0 |
| 9 | 0 |
| 10 | 0 |
| 11 | 0 |
| 12 | 0 |
| 13 | 0 |
| 14 | 0 |
| 15 | 0 |
| 16 | 0 |
| 17 | 0 |
| 18 | 0 |
| 19 | 0 |
| 20 | 0 |
| 21 | 0 |
| 22 | 0 |
| 23 | 0 |
| 24 | 0 |
| 25 | 0 |
| 26 | 0 |
| 27 | 0 |
| 28 | 0 |
| 29 | 0 |
| 30 | 0 |
| 31 | 0 |
| 32 | 0 |
| 33 | 0 |

Import

Export



Import requires the device to be disabled.

Teach

Velocity:

Start Teach



The following conditions are required before teaching:

- The device must be enabled.
- The device must be in velocity mode.
- The device must be in service mode.
- A move cannot be in progress.

The compensation table is displayed on the left. On the bottom there are buttons for importing and exporting the table from/to a csv file. Importing only works while the drive is disabled.

Controls for automatic teaching of the compensation table are displayed on the right.

12.5.1.3 Teaching

The **Start Teach** button, or the command `COGCOMP.TEACH` (→ p. 326), can be used to let the drive automatically detect cogging effects present and generate a compensation table.

The following prerequisites have to be met before starting:

- The drive has to be enabled and in Service mode and Velocity mode.
- When using a non-absolute feedback a homing has been executed.
- No other motion can be active.
- If on a linear axis, `COGCOMP.RANGEHIGH` (→ p. 323) and `COGCOMP.RANGELOW` (→ p. 324) have to be set to specify a range in that cogging compensation should be applied.

The current position and the Velocity set in COGCOMP.V (→ p. 327) have to allow the motor to completely move through the range.

When all prerequisites are met and the teaching is started the motor will move at the specified speed until it has either moved approximately 2 revolutions using a rotary motor, or moved through the whole specified range using a linear motor. If cogging compensation was enabled, it will automatically be disabled.

If the teaching was started using the button in the Cogging Compensation view, the table will be saved to non-volatile memory and be restored after power cycling the drive. Otherwise, the table can be saved manually using COGCOMP.SAVE (→ p. 325).

Should the motion during the teaching be stopped, either by the user or by the drive (e.g. by activating a limit switch), the teaching will be aborted, all correction values will be set to zero, and a warning will be displayed.

⚠ WARNING

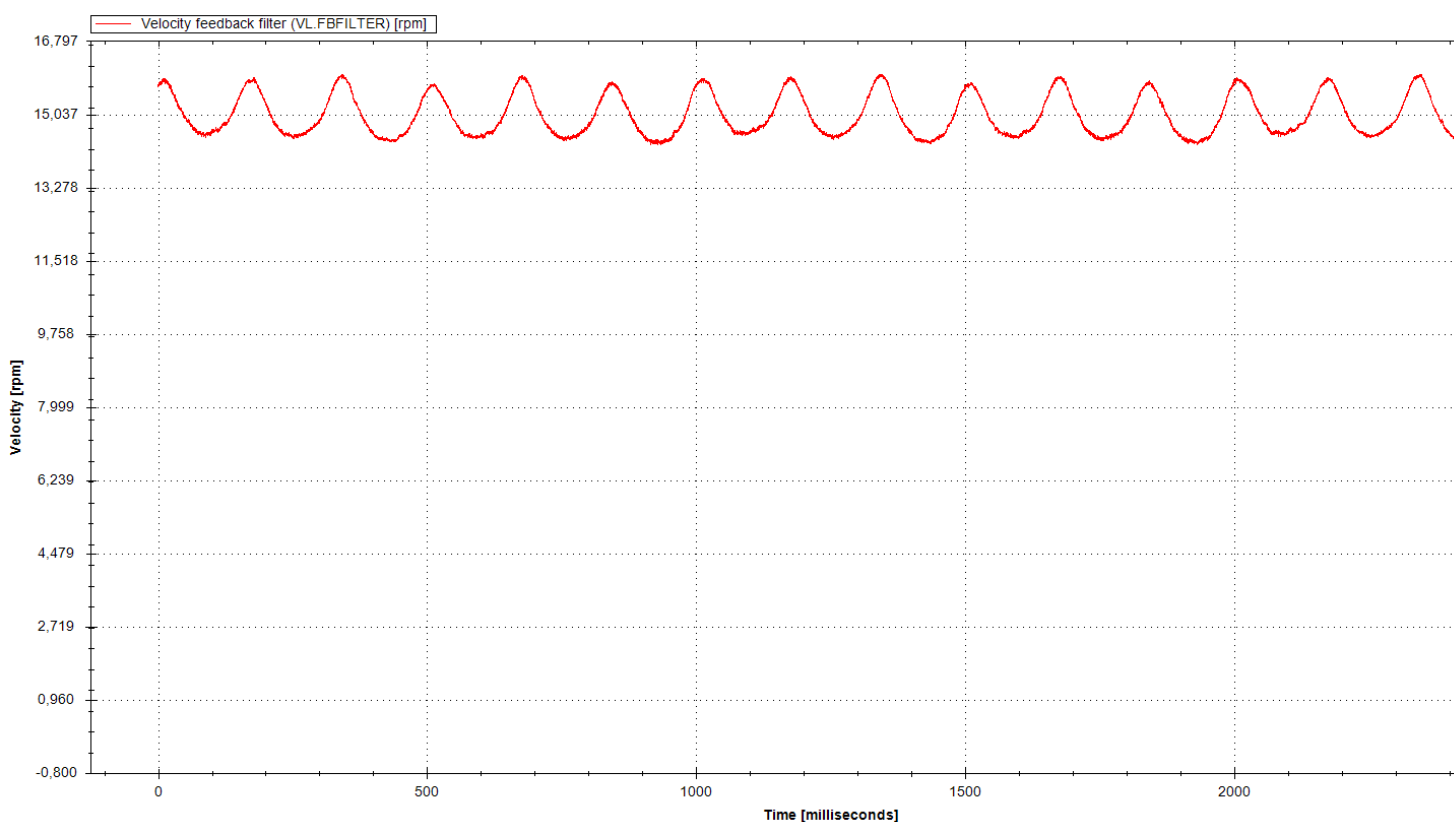
Compensation values are tied to the position feedback of the drive. Adding or removing any offsets, homing to a different position, changing the Cogging Compensation Range or otherwise influencing the position feedback may lead to the Compensation values not matching the cogging present in the motor and thus lead to unexpected motion.

12.5.1.4 Validating Cogging Compensation

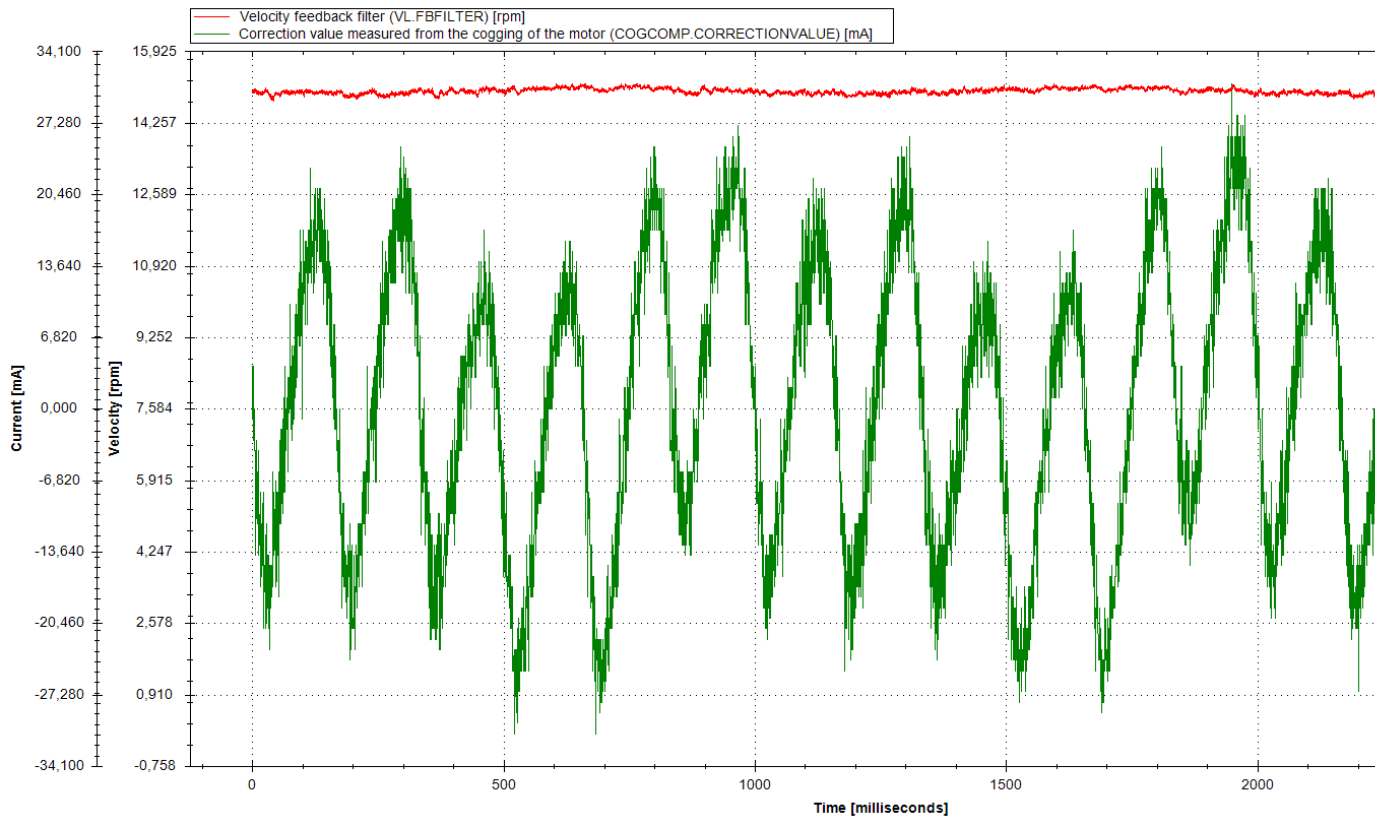
To validate the effect of cogging compensation, COGCOMP.CORRECTIONVALUE (→ p. 320) can be recorded in a [scope](#). It shows the feedforward value that is currently applied to compensate cogging.

The following two images show the effect on the velocity in two scopes, the first without compensation and the second with compensation enabled and COGCOMP.CORRECTIONVALUE scoped in addition to the velocity. The same motor and tuning was used and the same motion of constant 15 RPM was commanded. The cogging compensation table used was created using automatic teaching.

Cogging compensation disabled:



Cogging compensation enabled:



13 Scope

13.1 Overview

The scope allows you to plot up to six different parameters from the drive. Use **Full View** and **Normal View** to toggle between the scope setup (normal) and a larger view of only the scope output (full). You can configure, save, and restore scope settings from the normal view. The lower right corner of the normal view also includes a box that indicates status, drive and scope control buttons (**Stop Motion**, **Enable Drive**, **Start Recording**, and **Refresh**).

13.2 Using the Scope

You can set up scope plots using the tabs summarized below:

| Tab | Function |
|---------------------------------------|--|
| Channels | Select data source, plot axes, and plot appearance. |
| Time Base and Trigger | Select how much data to record and when to start recording the data. |
| User Traces | Defines traces to calculate based on two existing channels. |
| Servo Gains | Adjust the servo loop gains. |
| All Gains | View all current tuning gains in the drive and manually edit gains. |
| AR1, AR2, AR3, AR4 | Adjust filter settings. |
| Save and Print | Save the plot as a raw data file or as an image file; email the plot; print the plot; open the data file in Excel. |
| Measure | Display basic data read from the plots. |
| Cursors | Turn on the cursors and view the data at the cursor positions. |
| Display | Pan, zoom, and control the grid and background color. |
| Settings | Configure presets and y-axis group settings. |

13.2.1 Scope Channels Tab

The **Channels** tab allows you to select and record up to six channels simultaneously. Select the data to record for each channel from the lists in the **Source**, **Color**, **Y-axis**, and **Filter** and **Filter Frequency** columns. Once a recording is shown on the scope screen, you can click **Hide** to remove a channel from the scope display.

| Channels | | | | | | |
|----------|-----------------------------|--------|--------------------------|----------|--------------------------|------------------|
| Id | Source | Color | Hide | Y Axis | Filter | Filter Frequency |
| 1 | Current feedback (IL.FB) | Red | <input type="checkbox"/> | Current | <input type="checkbox"/> | 400 |
| 2 | Velocity command (VL.CM...) | Green | <input type="checkbox"/> | Velocity | <input type="checkbox"/> | 400 |
| 3 | Velocity feedback (VL.FB) | Blue | <input type="checkbox"/> | Velocity | <input type="checkbox"/> | 400 |
| 4 | None | Pink | <input type="checkbox"/> | Default | <input type="checkbox"/> | 400 |
| 5 | None | Purple | <input type="checkbox"/> | Default | <input type="checkbox"/> | 400 |
| 6 | None | Orange | <input type="checkbox"/> | Default | <input type="checkbox"/> | 400 |

13.2.1.1 Source Column

To set a channel to record, click the source you want to set and choose the appropriate channel. You can choose from None (no data is collected on that channel), preset trace types, or enter a user defined trace. Choosing “<User Defined>” allows you to record data from pre-defined locations. These locations are provided by the factory to collect less common values.

| Channels | | | | | | |
|-----------------------|---------------------------------|--------|--------------------------|------------|--------------------------|------------------|
| Time-base and Trigger | | | | | | |
| Service Motion | | | | | | |
| Servo Gains | | | | | | |
| Observer | | | | | | |
| All Gains | | | | | | |
| Id | Source | Color | Hide | Y Axis | Filter | Filter Frequency |
| 1 | Analog input signal (AIN.V) | Red | <input type="checkbox"/> | Analog I/O | <input type="checkbox"/> | 400 |
| 2 | Analog input signal (AIN.V) | Green | <input type="checkbox"/> | Velocity | <input type="checkbox"/> | 400 |
| 3 | Analog output user value (AOUT) | Blue | <input type="checkbox"/> | Velocity | <input type="checkbox"/> | 400 |
| 4 | Analog output value (AOUT) | Purple | <input type="checkbox"/> | Default | <input type="checkbox"/> | 400 |
| 5 | Bus voltage (VBUS.VALUE) | Pink | <input type="checkbox"/> | Default | <input type="checkbox"/> | 400 |
| 6 | Current - U winding of motor | Purple | <input type="checkbox"/> | Default | <input type="checkbox"/> | 400 |
| | Current - V winding of motor | Orange | <input type="checkbox"/> | Default | <input type="checkbox"/> | 400 |
| | Current command (IL.CMD) | Orange | <input type="checkbox"/> | Default | <input type="checkbox"/> | 400 |

13.2.1.2 Color Column

For valid sources, you can click on the color in the Color tab and choose a different color than the default, or create a custom color.

13.2.1.3 Hide Column

You can check the **Hide** box to hide a given plot trace. This feature can make it easier to focus on specific data as needed.

13.2.1.4 Y-Axis Column

The Y-axis column allows you to choose on which Y-axis the channel will be displayed. Several predefined Y-axis groups exists. Click on the item in the column to change the label for the trace.

13.2.1.5 Filter and Filter Frequency Column

Check this box and use the frequency column to apply a low pass filter to the data collected. The filter is applied when the data is collected. It is not applied to data already collected if this is checked after the data is collected.

13.2.2 Measure Tab

The measure tab displays basic data reads from the plots.

13.2.2.1 AC RMS and True RMS

In the measure tab, there is a column for AC RMS and a column for True RMS. True or full RMS is the full heating value of a signal and includes any DC terms in the value. AC RMS removes any DC value and gives only the RMS value as a measure of a signal's standard deviation.

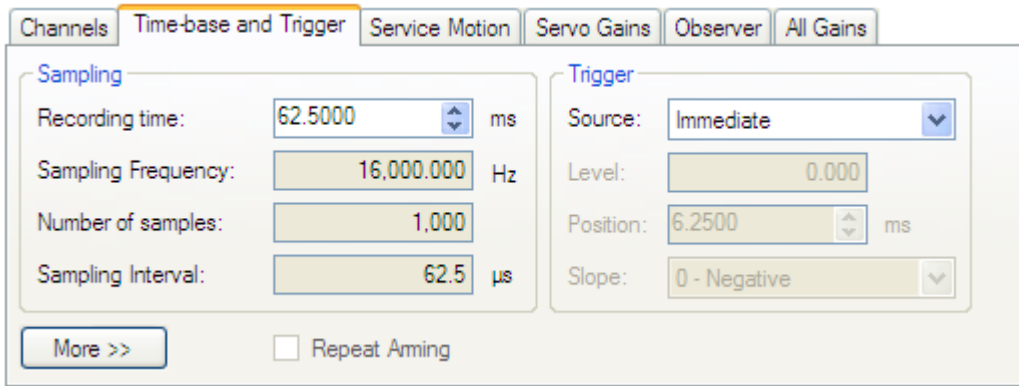
$$\text{True RMS} = \sqrt{\text{Sum}(x[n]^2)/N} \text{ where } N \text{ is number of points}$$

$$\text{AC RMS} = \sqrt{(\text{True RMS})^2 - (\text{dc or average value})^2}$$

13.2.3 Scope Time-base and Trigger Tab

Use the **Time-base and Trigger** tab to select how much data to record and when to start (trigger) recording the data. You can set length of recording in ms and the sampling frequency in Hz. The number of samples is a calculated value displayed for reference. The trigger can be set to trigger immediately when you click **Start Recording** or to trigger when a specified value for a given signal is reached. The default **Time-base and**

Trigger view specifies recording time, sampling frequency, and either an immediate trigger or a trigger based on a specified signal. Click the **More** button in this view to specify a given number of samples, sampling frequency, sampling interval, and access additional trigger options.

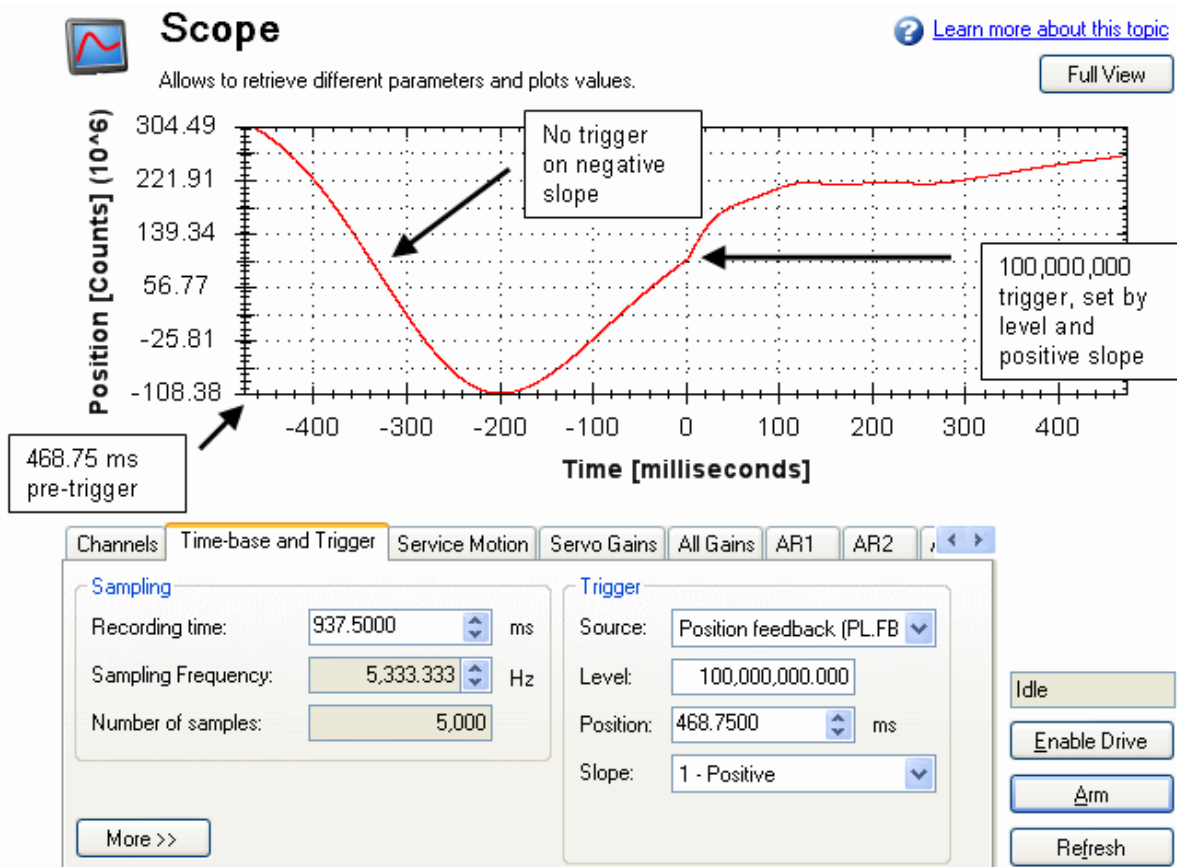


In this tab, you can set length of recording in ms and the sampling frequency in Hz. The number of samples is a calculated value displayed for reference. You can also choose the trigger source to be **Immediate** (triggers as soon as you click the **Start Recording** button) or to be one of many predefined sources.

If you choose a source other than **Immediate**, you can set the level, position, and slope for the trigger value.

- Level sets the value of the source that triggers the recording to start.
- Position sets the amount of time that the scope displays before the trigger occurred.
- Slope sets whether the source data must pass the level value in a positive or negative direction.

An example of triggering is shown below:



13.2.3.1 Scope Time-base and Trigger, More View

Click **More** to display additional options for configuring the time-base and trigger.

| Channels | Time-base and Trigger | Service Motion | Servo Gains | Observer | All Gains |
|--|-----------------------|--|-------------|----------|-----------|
| Sampling Recording time: <input type="text" value="62.5000"/> ms Sampling Frequency: <input type="text" value="16.000.000"/> Hz Number of samples: <input type="text" value="1.000"/> Sampling Interval: <input type="text" value="62.5"/> μ s <input type="button" value="Less <<"/> <input type="checkbox"/> Repeat Arming | | Trigger Type: <input type="text" value="0 - Immediate"/> Source: <input type="text" value="IL.FB"/> Level: <input type="text" value="0.000"/> Position: <input type="text" value="10"/> % Slope: <input type="text" value="0 - Negative"/> | | | |

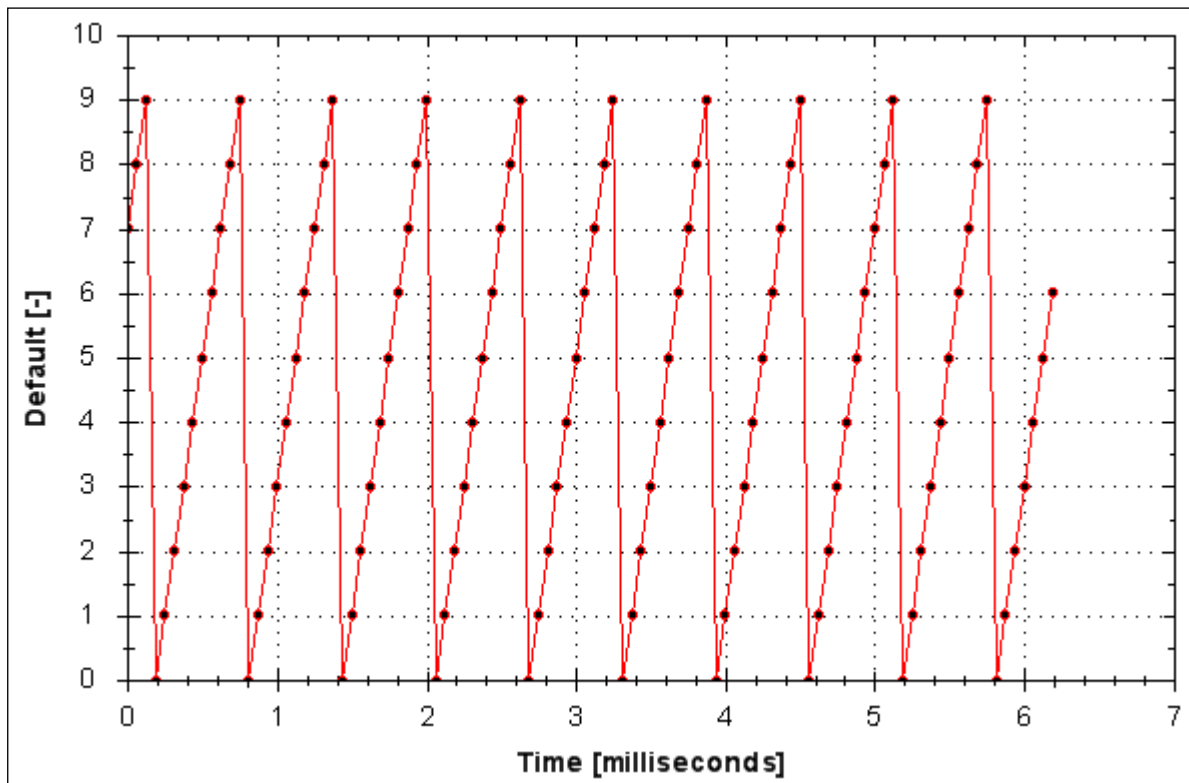
In the **Sampling** area of this view, you can specify the recording length by entering a sampling frequency and a number of samples. Here, the recording time is a calculated value displayed for reference.

What is triggering?

Triggering allows you to precisely control the start point of data collected in the scope. For example, if you are looking for a large spike, you can set the trigger to start the scope to begin recording when it sees the large spike. This section describes the triggering functionality of the scope.

Test Signal

As an example, it is useful to examine variations on a record of a test signal that generates a sawtooth signal. The signal starts at 0 and increases by one every drive sample (1/16,000 second) to a maximum of 9, and then returns to 0. This signal continues indefinitely. The record of this signal is shown below.



13.2.3.2 Trigger Type

The **Trigger** area in the **More** view offers more flexibility than the default view. You can specify four types of trigger types (REC.TRIGTYPE):

- **Immediate - 0.** This mode will start recording as soon as the recording command (REC.TRIG) is received by the drive.
- **Command / On Next Command – 1.** This trigger type lets you specify a trigger on the next telnet command received by the drive. This is useful in a telnet session via Hyperterminal (or a similar program). EWV¹ is constantly sending telnet commands, so this is not typically used in a EWV session.
- **Parameter / On Source Signal – 2.** This trigger type lets you specify a trigger source and set of conditions to trigger recording of data. This is very similar to the triggering used on oscilloscopes.
- **Boolean – 3.** This trigger type lets you trigger on a boolean (0 or 1), such as drive active status.

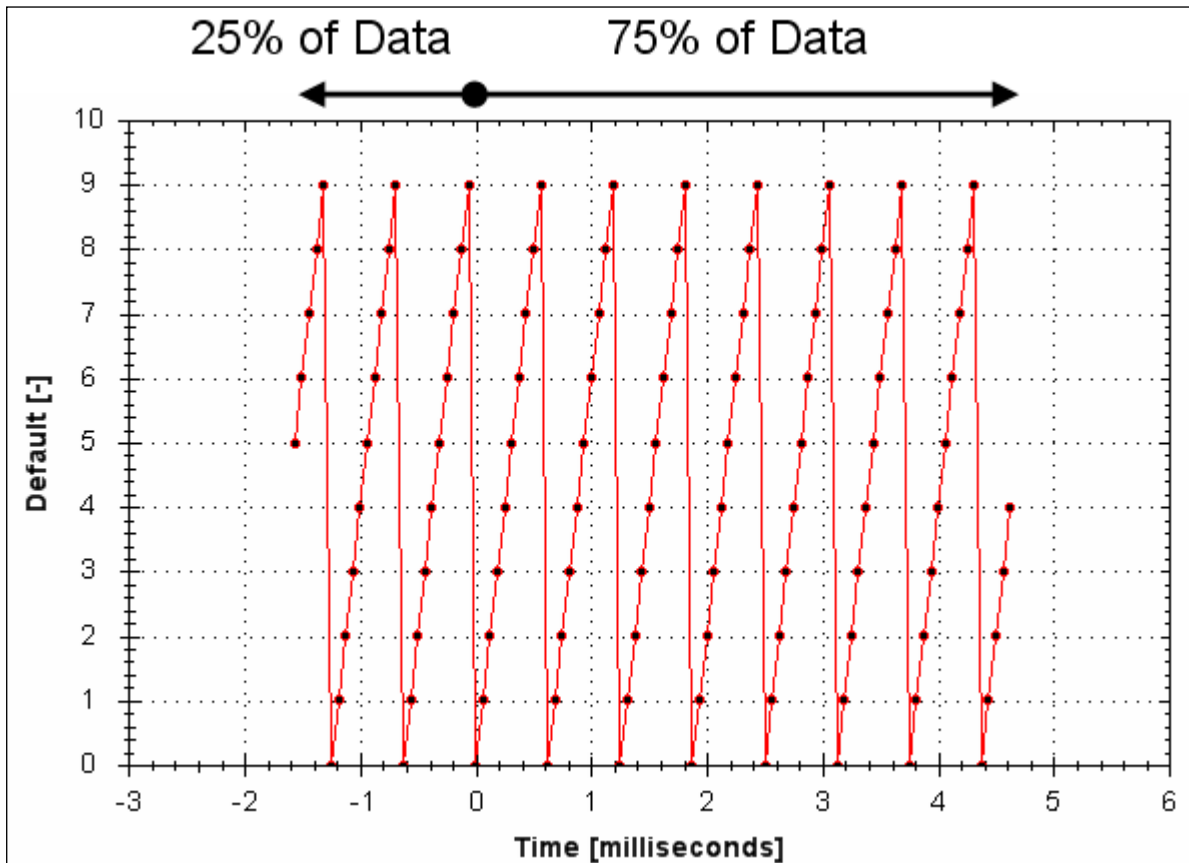
13.2.3.3 Trigger Position

Trigger Position (REC.TRIGPOS) allows you to collect data that occurs before the trigger occurs. If you have a rare condition, you may want to see the conditions that led up to it. Trigger position lets you control how much signal is collected before the trigger condition occurred.

Trigger position is specified in units of percent (%). If you specify a trigger position of X% , X% of the data is before 0 ms in the data time and 100-X% (the rest of the data) is at or greater than 0 ms. In the picture below, trigger position is set to 25% (REC.TRIGPOS 25).

In the EWV scope, the 0 time point is clear. When collecting the data via REC.RETRIEVE or similar commands, the time is not returned, so some caution should be used when the trigger point is important to understand.

Trigger position is not used in trigger type “Immediate” (TRIGTYPE 0).



13.2.3.4 Trigger Value

The trigger value (REC.TRIGVAL) specifies a target value that should trigger the recording to start. The trigger value is used in trigger type Parameter / On Next Signal only.

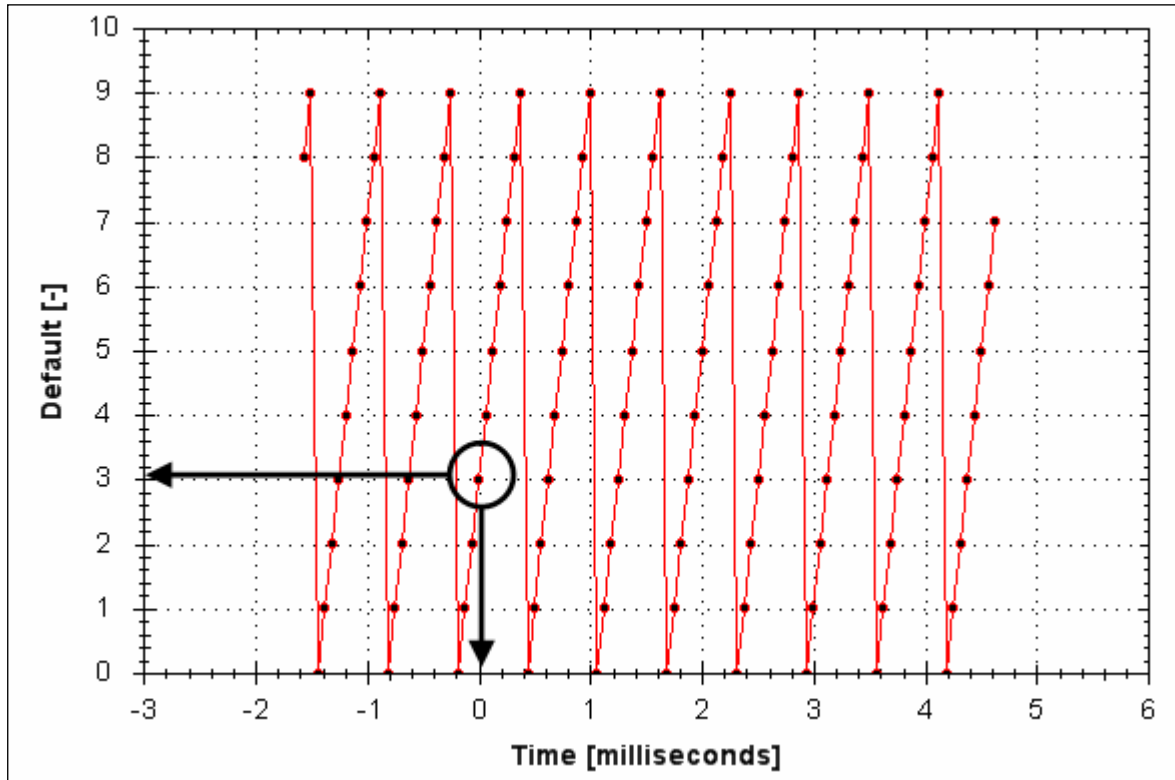
¹Embedded Workbench Views

The trigger value is not used in the boolean trigger type. Use the trigger slope to set the polarity of the boolean trigger.

When the trigger slope is positive, the trigger value will trigger when:

- The trigger source is less than the trigger value in the previous recording sample
- The trigger source is greater than or equal to the trigger value in the current recording sample

Below is an example showing triggering of trigger value of 3 (REC.TRIGVAL 3) and positive trigger slope (REC.TRIGSLOPE 1). You can see that the recording triggers at time zero when the source reaches the value of 3.



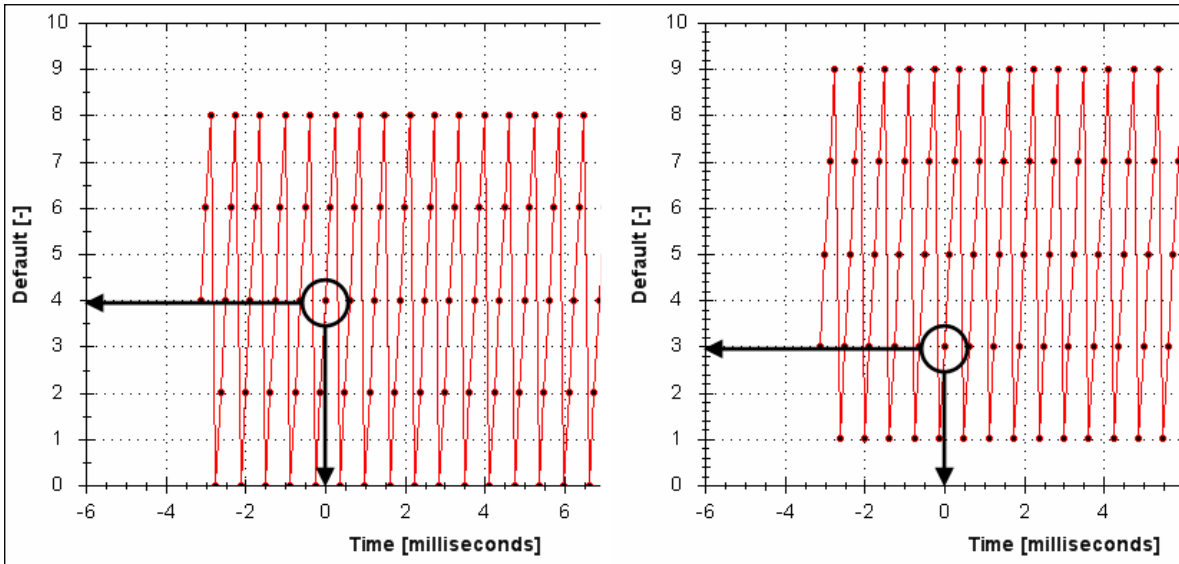
When the trigger slope is negative, the trigger value will trigger when:

- The trigger source is greater than the trigger value in the previous recording sample.
- The trigger source is less than or equal to the trigger value in the current recording sample.

13.2.3.5 Effects of Recorder Gap

When the recording rate is less than 16,000 Hz (REC.GAP > 1), there can be some impact on the triggering of the recorder. When using pretriggering and a recording rate of less than 16,000 Hz, the trigger only evaluates every N samples, where N is the value of REC.GAP. Two effects result from this condition:

1. You cannot be sure of the moment that the recorder is triggered any closer than N samples. An example of this is shown below where the trigger value is set to 3, the trigger slope is positive and the recorder gap is 2. Both examples are the same data, but one instance collected and triggered on the odd data. The other example collected and triggered on the even data.



2. You can miss triggers, whose duration is less than N samples, where N is the value of REC.GAP. This is because the trigger is only evaluated every N samples.

A workaround for the above effects is available by setting the recorder trigger position to zero (REC.TRIGPOS 0). This eliminates conflicts between pretrigger and post-trigger timing and will guarantee trigger evaluation every sample, eliminating the cases above.

13.2.3.6 Trigger Slope

Trigger Slope specifies whether you trigger on a positive or negative change in the trigger source. The effect of the trigger slope is different for trigger type Boolean and On Next Signal modes.

Boolean Trigger Type

When using Boolean type:

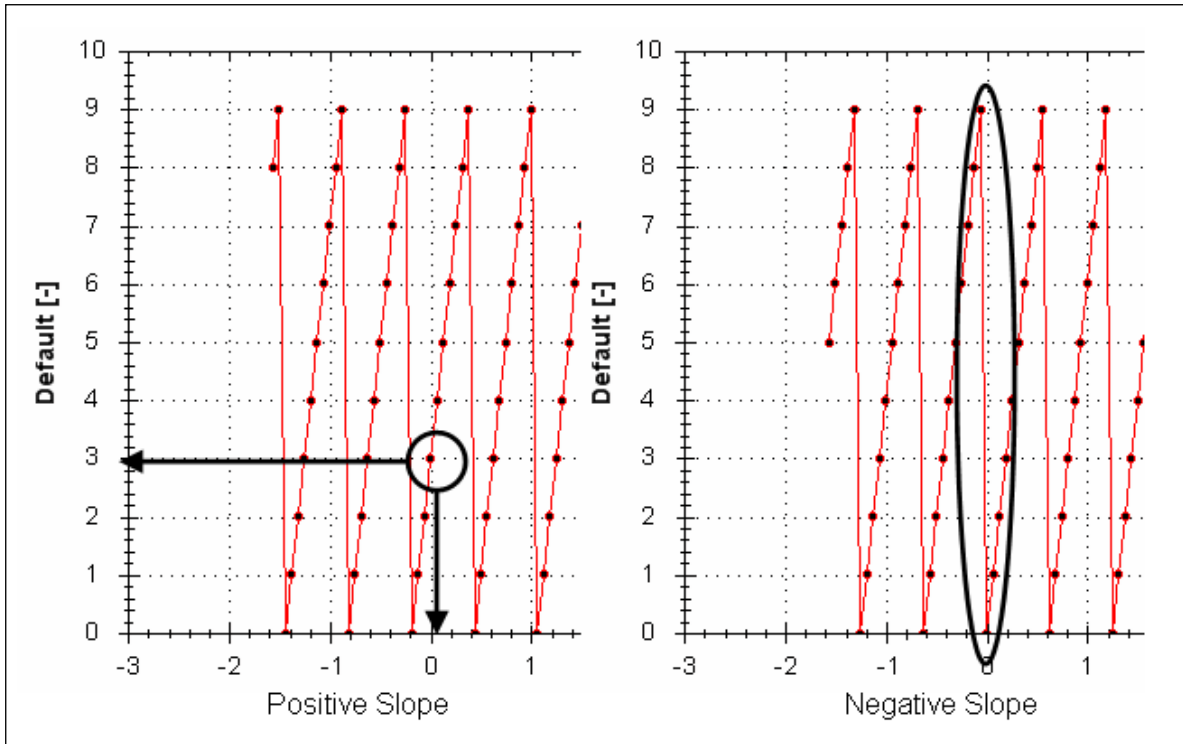
- A positive slope will trigger when the trigger source is 1
- A negative slope will trigger when the trigger source is 0

The boolean trigger type is a state trigger. There is no need to transition from 0 to 1 to trigger with the positive slope. If the trigger source is 1 from the start, the positive slope will immediately trigger.

On Next Signal Trigger Type

The “On Next Signal” trigger type allows you to specify if the recorder should trigger when the signal crosses the trigger level in the positive or negative direction. The signal only needs to reach the trigger level; it does not need to pass the trigger level.

In the examples below, the trigger value is set to 3 (REC.TRIGVAL 3.000). You can see that with positive slope, the trigger occurs exactly when the signal transitions from 2 to 3, because it reached 3. The negative slope case triggers when the signal transitions from 9 to 0, because it crossed 3 on the way.

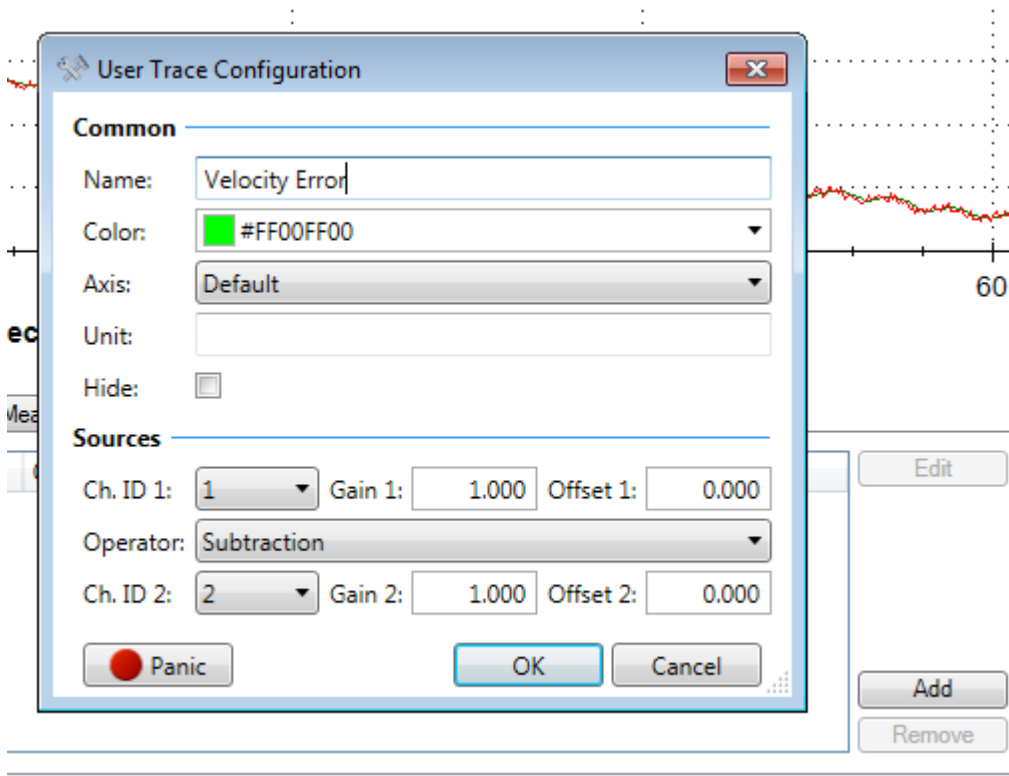


13.3 User Traces

The User Traces tab is used to create a custom trace from the data of two other channels. A user trace can multiply, divide, add, or subtract two channels.

13.3.1 Add and Configure a User Trace

From the User Traces tab, select Add.



The user trace configuration uses two channels that have already been defined in the Channels tab, and performs a calculation on them. To change which parameter sources are being used for the calculation, either navigate to the Channels tab, and choose a new channel source, or point your user trace Ch. IDs to use a different existing channel.

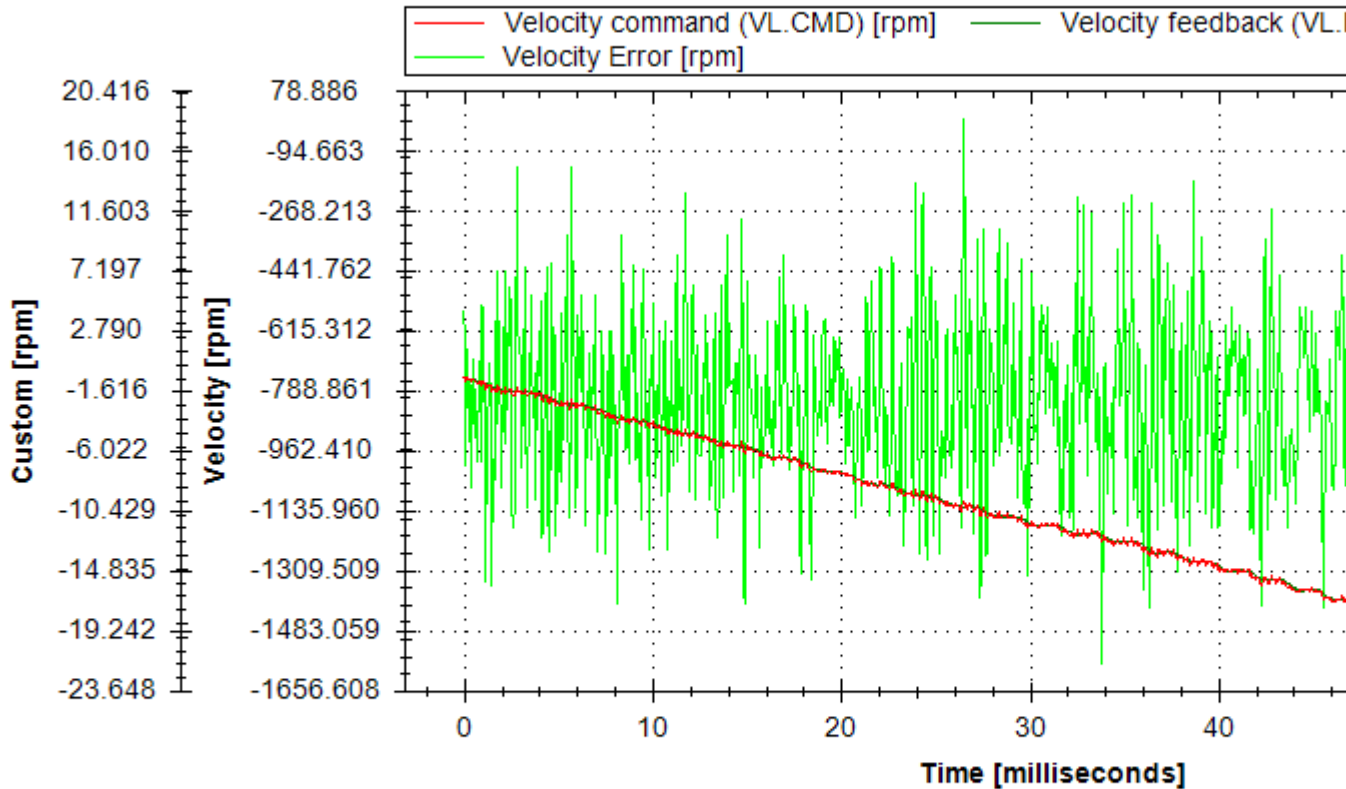
| Setting | Description |
|----------|---|
| Name | Name to display in the scope for this trace. |
| Color | Color of the trace in the scope. |
| Axis | Defines the y-axis the trace is drawn on. Choose from an existing axis or from a custom axis created from the Settings tab. |
| Unit | Custom string defined by user. This string is only for user tracking and has no effect on any units. |
| Hide | Check to hide this trace from the scope. |
| Ch. ID 1 | Select the first channel whose data will be used in the calculation. The gain and offset are also configurable and will be calculated as $gain1 * (Ch. ID1 + offset1)$. |
| Operator | Choose the type of calculation to be performed. |
| Ch. ID 2 | Select the second channel whose data will be used in the calculation. The gain and offset are also configurable and will be calculated as $gain2 * (Ch. ID2 + offset2)$. |

Select Ok after configuring, and the custom trace will be added to the scope. Defining a custom y-axis (in the Settings tab) allows the trace to be shown on a different scale. In the example below, Velocity Error is a User trace of VL.CMD - VL.FB shown on the axis named Custom, while VL.CMD and VL.FB are shown on the Velocity axis.



Scope

Allows to retrieve different parameters and plots values.



| Name | Color | Hide | Axis | Unit | Source ID 1 | Gain 1 | Offset 1 | Operator | Source ID 2 | Gain 2 | Offset 2 |
|---------------|--------------------------------------|--------------------------|--------|------|-------------|--------|----------|-------------|-------------|--------|----------|
| Velocity E... | █ | <input type="checkbox"/> | Custom | rpm | 1 | 1.000 | 0.000 | Subtraction | 2 | 1.000 | 0.000 |

If a User Trace points to invalid data, it will not draw the data in the scope and a warning sign will appear. Check your Ch. IDs of your User Trace and the channel sources in the Channels tab to make sure both are properly configured.

| Name | Color | Hide | Axis |
|----------------|--------------------------------------|--------------------------|--------|
| Velocity Error | █ | <input type="checkbox"/> | Custom |

13.3.2 Saving User Traces

User traces can be saved for later use when [creating a new preset](#).

NOTE

User traces will not be saved when saving a csv file from the Save and Print tab. Only the channels from the Channels tab will be saved to the file.

13.4 Scope Settings

Scope settings are used to store and retrieve the scope parameters. You can save multiple settings, called "presets", under different names. You can save, delete, import, or export the presets. The settings are stored in EWW project file (default.wbproj) and settings are common to all the drives in EWW.

NOTE

The Load at Startup option is enabled/disabled from the Tools > User Options menu in KAS.

13.4.1 Load a setting (preset) to Scope screen

In Scope Settings section, the existing presets are listed in the **Select Setting** box. To load a setting to the scope screen, select the desired preset from the **Select Setting** list.

13.4.2 Create a new preset

1. Modify any scope parameters.
2. Select the **Settings** tab.
3. Click **Save As**. The following dialog is displayed:

4. Enter the setting name and click **OK**. The current settings are saved as a preset with the given name and displayed in the list.

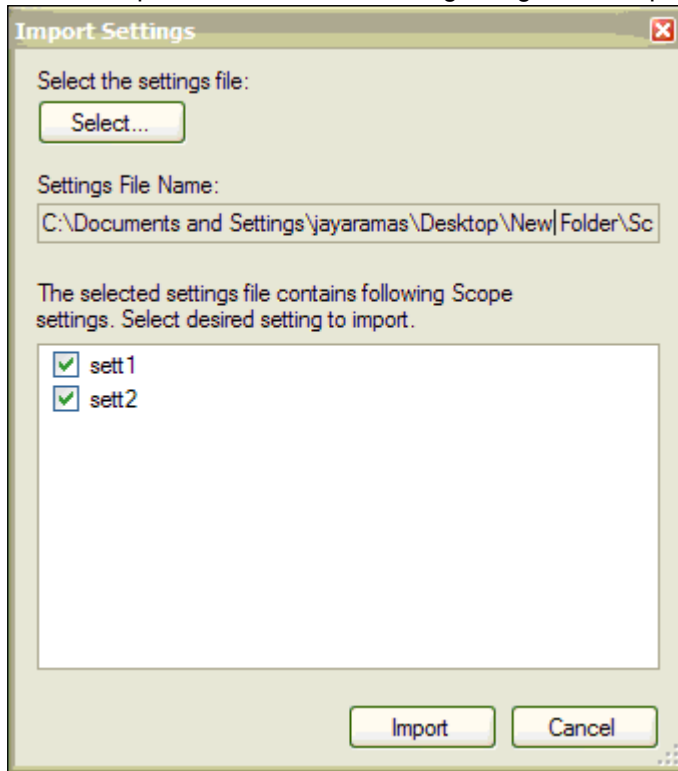
13.4.3 Save or delete preset

Save saves any modification to the open preset. **Delete** deletes the open preset.

13.4.4 Import preset

Import the presets contained in the selected settings file as follows:

1. Click on Import button and the following dialog will be displayed.

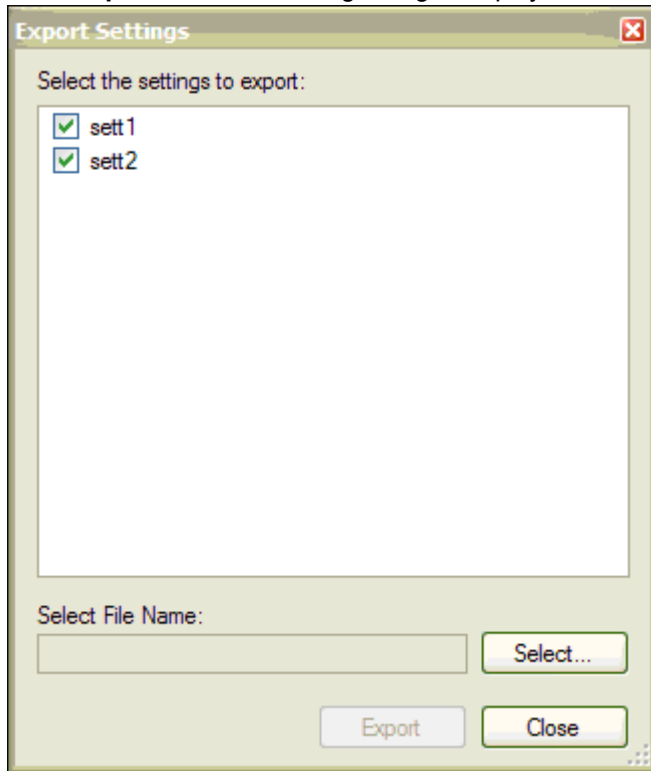


2. Select the settings file by clicking “Select...” button.
3. All the scope presets will be displayed contained in the selected settings file.
4. Select/Deselect the presets and then click on Import.
5. If preset name already exists in application the confirmation message will be shown to user to replace it or to ignore.

13.4.5 Export preset

Export a preset to a file as follows:

1. Click **Export** and the following dialog is displayed:



2. The existing presets are displayed and user can select/deselect the preset to export.
3. Select the file name to export.
4. Click **Export** to export the selected presets to a file.

13.4.6 Scope axis scaling and zooming

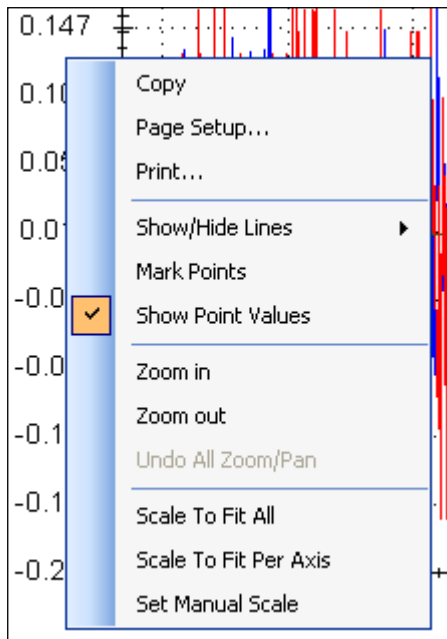
The scope provides two mechanisms for determining how you view the data:

- Scaling: you can choose the scale for the different axes.
- Zooming: you can choose a particular portion of the scope that you want to observe more in details, and then come back to previous scaling.

Two different scaling modes are provided on each axis:

- Manual: you can determine the minimum and maximum value of the axis (X or Y axis).
- Scale to fit: the program will compute a scale for this axis that will display all the curves bound to it (X or Y axis).

These functionalities are accessible through the contextual menu when right-clicking in the axis zone. A simple left-click in the axis zone will provide the manual range functionality. A supplementary functionality allows you to perform a scale to fit on all axes is also available, which allows a good overview.

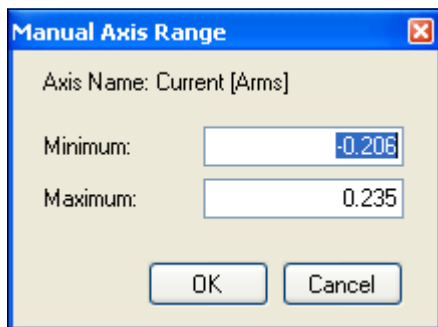


The zoom functionality allows you to navigate in a portion of the graphic. When you reset the zoom, the initial scales are shown.

In the display tab, when “Remember Axis Scale” is set, the scales of the axes are kept between two sequential recordings. You can fine tune the scale to visualize a particular behavior and record a second time and see the same behaviour without having to redo all the tuning. When not checked, a scale to fit all will be performed after each record. This setting is reset when exiting EWV and should be explicitly set at next startup.

13.4.7 Manual range per axis

After recording data, right click anywhere on the y-axis and select **Set Manual Scale** to open a dialog box to set the range for the axis. Enter the Y-axis minimum value and Y-axis maximum value. Click **OK** to reset the Y-axis to new range.



13.4.8 Unit display on Y axis

The unit on the Y-axis is displayed if all scope signals units are identical for that Y-axis. If different units apply to different signals, the units are displayed as [-]. For example, if the velocity Y-axis has signals VL.FB and IL.CMD, then the unit displayed is [-], since the units for these parameters are different. If IL.CMD is hidden, then the correct unit for VL.FB, rpm, is displayed.

Related Parameters:

BODE Parameters (→ p. 281)

14 Using Parameters and the Terminal Screen

| | |
|---------------------------------------|------------|
| 14.1 Terminal | 169 |
| 14.2 Parameter List | 173 |
| 14.3 Parameter Load/Save | 175 |

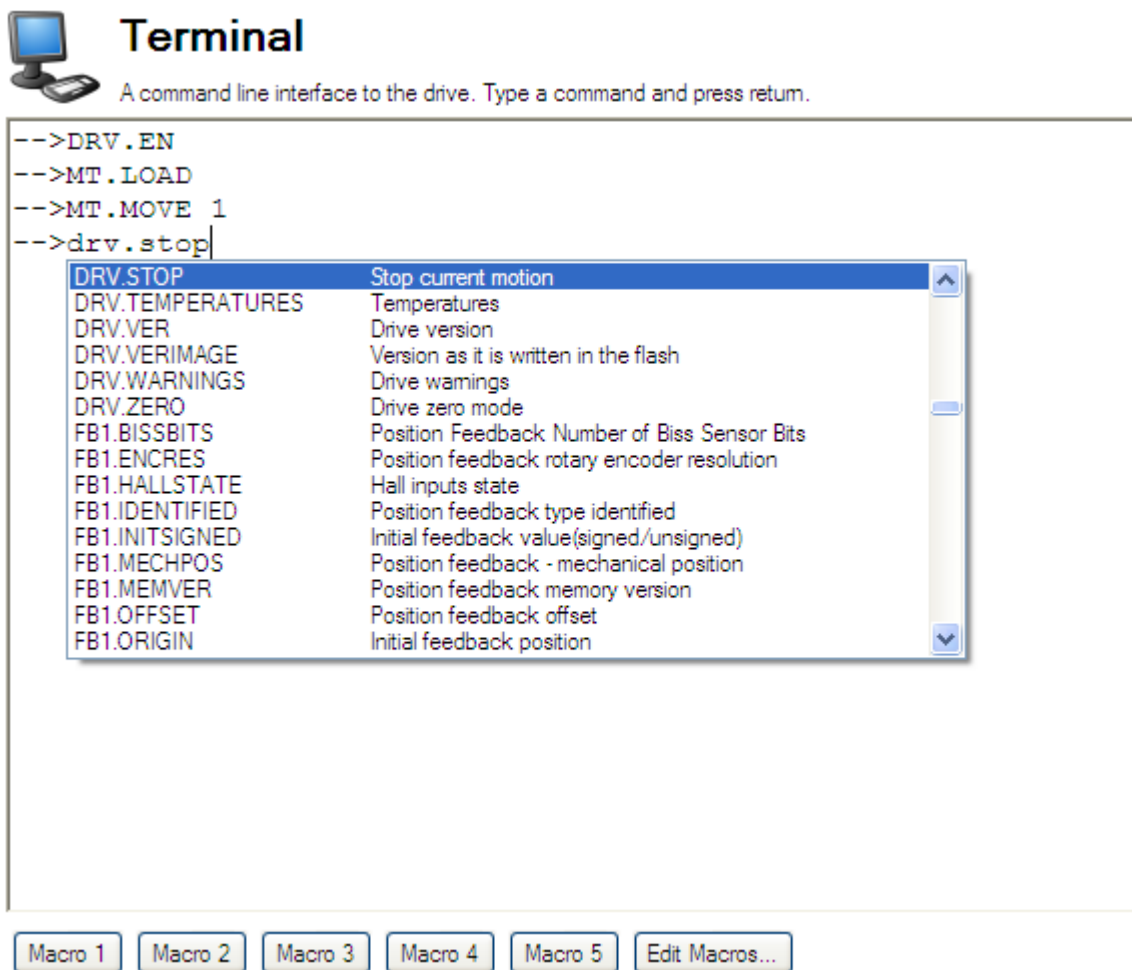
14.1 Terminal

14.1.1 Overview

The terminal mode provides a quick and easy way to issue commands to the drive directly. Typically used by "power users" who are familiar with the command set, the terminal mode can help in setup, troubleshooting, and other diagnostic actions. When using the terminal mode, [E_{WV}¹](#) shows the parameter and command set in a popup view and uses an autocomplete tool to help you select the proper parameter or command. Right-click in the command entry area to open a popup menu for editing commands and for clearing the screen. The terminal also provides a macro editor that allows a series of commands to be executed via a single command (called a macro). Macros are useful when you must frequently execute a sequence of commands.

14.1.2 Using the Terminal

Click Terminal in the navigation tree to issue parameters and commands in the terminal mode. You can enter parameters and commands at the prompt as shown in this example:



Terminal
A command line interface to the drive. Type a command and press return.

```
-->DRV.EN
-->MT.LOAD
-->MT.MOVE 1
-->drv.stop|
```

| | |
|------------------|--|
| DRV.STOP | Stop current motion |
| DRV.TEMPERATURES | Temperatures |
| DRV.VER | Drive version |
| DRV.VERIMAGE | Version as it is written in the flash |
| DRV.WARNINGS | Drive warnings |
| DRV.ZERO | Drive zero mode |
| FB1.BISSBITS | Position Feedback Number of Biss Sensor Bits |
| FB1.ENCRESP | Position feedback rotary encoder resolution |
| FB1.HALLSTATE | Hall inputs state |
| FB1.IDENTIFIED | Position feedback type identified |
| FB1.INITSIGNED | Initial feedback value(signed/unsigned) |
| FB1.MECHPOS | Position feedback - mechanical position |
| FB1.MEMVER | Position feedback memory version |
| FB1.OFFSET | Position feedback offset |
| FB1.ORIGIN | Initial feedback position |

Macro 1 Macro 2 Macro 3 Macro 4 Macro 5 Edit Macros...

The terminal supports the following keyboard shortcuts:

| Keyboard Shortcut | Description |
|-------------------|----------------------------|
| F2 | Executes the last command. |

¹Embedded Workbench Views

| Keyboard Shortcut | Description |
|-------------------|--|
| Up Arrow | Gets the previous command from the command history. |
| Down Arrow | Gets the next command from the command history. |
| CTRL+J | Shows the list of commands that the drive supports. |
| ESC | Hides the command list if command list is open. Clears the line if command list is not open. |
| Enter (Return) | Executes the current command. This command sends the text you have typed to the drive and then prints the text the drive returns on the next line. |

14.1.3 Macros

Macros are a short sequence of instructions that can be sent to the drive using a single button. You can create up to five macros within the terminal mode using the macro editor. Each of the macros can be given a name and will appear as a button below the Terminal screen area. When you select the button, the command set tied to that macro will appear in the Terminal area and the commands will be executed. If there are commands which are not entered correctly, the terminal screen area will indicate the errors.

You can access macros and the macro editor from the buttons are located at the bottom of the terminal screen. Click a macro button to execute the associated macro. The **Edit Macros** button opens the macro editor used to create and edit macros. You can also assign custom names to the macro buttons with this editor.

14.1.3.0.1 Creating a Macro from Terminal commands

You can create a macro from a sequence of terminal commands as follows:

1. Enter the sequence of commands.



Terminal

A command line interface to the drive. Type a command and press return.

```
Motion 1-->DRV.EN  
Motion 1-->MT.LOAD  
Motion 1-->MT.MOVE 1  
-->
```

Motion 1

Macro 2

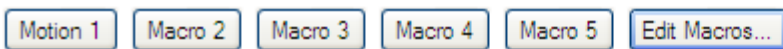
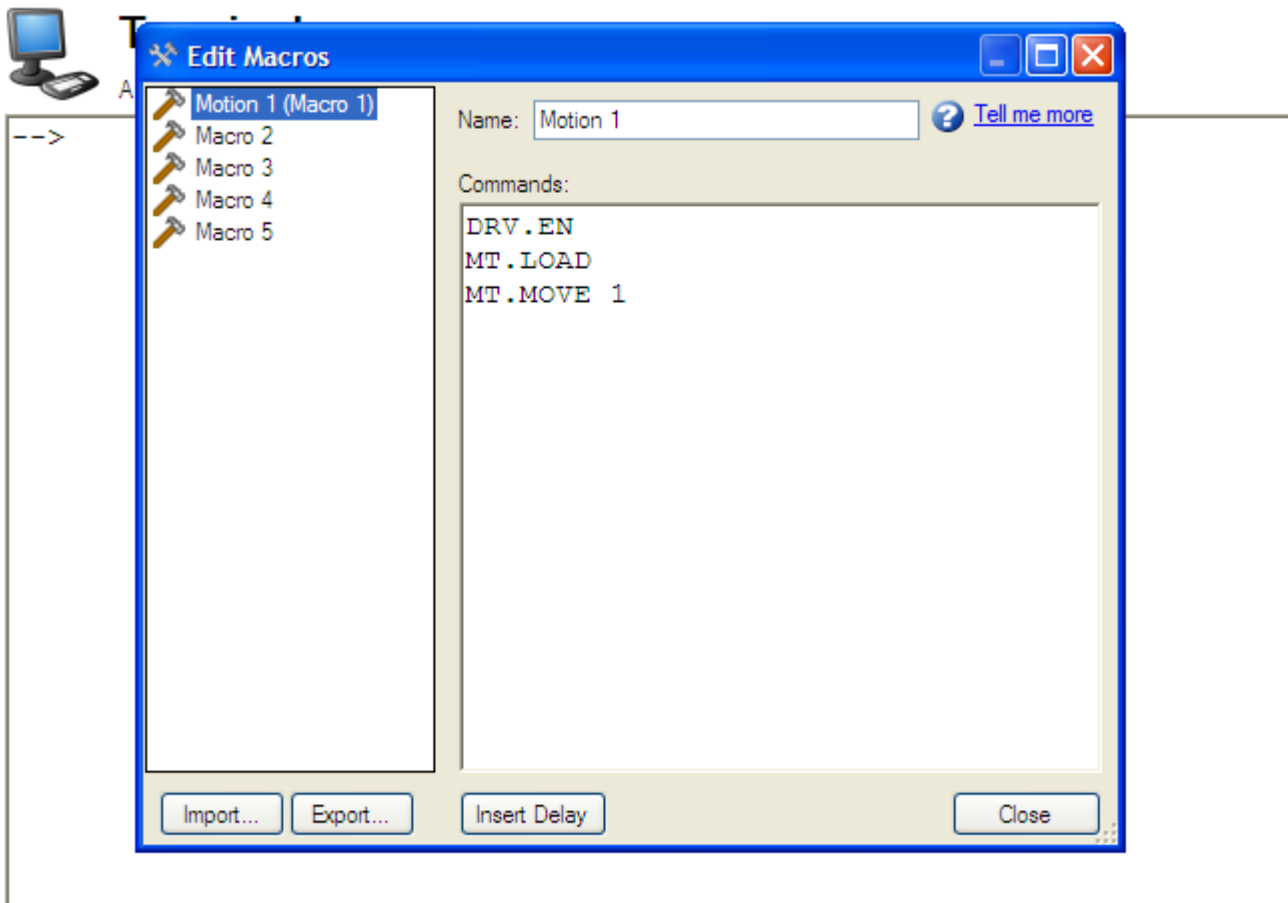
Macro 3

Macro 4

Macro 5

Edit Macros...

2. Copy the sequence of commands, then click **Edit Macro**. Select a macro from the tree on the left, then paste the sequence of commands into the Commands area of the Edit Macros window.



3. Select **Save** to save your macro.

14.1.3.0.2 Macro Editor

The macro editor allows you to create and modify up to five macros. EWV automatically saves the macros that you create. If you exit KAS IDE and then start KAS IDE again, the macros you defined will still be available.

To create or edit a macro from the editor, open the **Terminal** screen and click on **Edit Macros** (located at the bottom of the screen). Select the macro you wish to edit from the tree on the left, then use the features described below to build the macro. You can add comments in the macro after the ";" symbol.

| Button or Dialog Box | Description |
|----------------------|---|
| Name | Allows you to give each macro a unique name. This name will be shown on the button in the terminal view. |
| Commands | Displays the commands that are sent to the drive when you use this macro. The response from the drive is displayed in the terminal. |
| Import | Overwrites the selected macro with the contents of a macro file exported from another computer. |
| Export | Sends the selected macro to a text file so that you can import into KAS IDE running on another computer. |

| Button or Dialog Box | Description |
|----------------------|---|
| Insert Delay | Inserts a step into the macro that causes a delay for a specified number of milliseconds before continuing. A line that starts with <code>#delay 5000</code> will pause the execution of the macro for 5000 milliseconds. |
| Close | Closes this window and returns to the terminal view. |

14.2 Parameter List

This screen displays a list of the current values of all the parameters that the drive supports. You can sort some of the rows by clicking the column headers.



Parameters

This page lists all the current values of all the drive parameters on the drive.

| Full Name | Value | Units | Parameter | Read/Write |
|---|--------------------|-------------|-------------------|------------|
| Analog Input | | | | |
| Analog input low pass filter cutoff freq... | 5,000.000 | Hz | AIN.CUTOFF | read-write |
| Analog input signal deadband | 0.000 | V | AIN.DEADBAND | read-write |
| Analog Input Deadband Mode | 0 - Deadband | | AIN.DEADBANDMO... | read-write |
| Analog input torque scale | 0.001 | Arms/V | AIN.ISCALE | read-write |
| Analog input mode | 1 - Command Source | | AIN.MODE | read-write |
| Analog input offset | 0.000 | V | AIN.OFFSET | read-write |
| Analog input position scale | 0.000 | Counts16... | AIN.PSCALE | read-write |
| Analog input signal | 0.000 | V | AIN.VALUE | read-only |
| Analog input velocity scale | 0.060 | rpm/V | AIN.VSCALE | read-write |
| Analog Output | | | | |
| Bode | | | | |
| Bus Volts | | | | |
| Controlled Stop | | | | |
| Current Loop | | | | |
| Digital Input | | | | |
| Digital Output | | | | |
| Drive | | | | |
| Electronic Gearing | | | | |
| Fieldbuses | | | | |
| GUI Parameters | | | | |
| Hardware Limit Switches | | | | |
| Homing | | | | |
| TCP/IP | | | | |
| Motion Tasks | | | | |
| Motor | | | | |
| Position Capture | | | | |
| Position Feedback | | | | |
| Position Loop | | | | |
| Programmable Limit Switches | | | | |
| Recorder | | | | |
| Regen Resistors | | | | |
| Service Motion | | | | |
| Software Limit Switches | | | | |
| Units | | | | |
| Velocity Loop | | | | |
| Wake & Shake | | | | |
| Misc | | | | |

| Button or Dialog Box | Description |
|----------------------|--|
| Refresh | Reads all the parameters from the drive and update the contents of the table. |
| Print | Sends the data that you see on the screen to the printer. |
| eMail | Opens an email message and attaches a comma separated file to the email message. |

| Button or Dialog Box | Description |
|----------------------|--|
| More/Less | Adds two more columns to the table with the range and default values for each parameter. |

14.3 Parameter Load/Save

To copy a configuration (all the device parameters that are stored in the device's nonvolatile memory) from one device to another, click the **Save to File** button on the first device and then use the **Load from File** on the second device. If you wish the second device to keep these new parameters after the device is turned off, then you can save the parameters to the nonvolatile memory with **Save To Device**.

Check **Add non-volatile save instructions to file** to append a DRV.NVSAVE to the end of the saved file. Any load of this file will automatically save the parameters in NV RAM.

Check **Save to non-volatile when loaded** to force an NV save after a file is loaded.

⚠ WARNING

When loading a parameter file to a new device, if MOTOR.AUTOSSET = 0 then motor parameters should be checked to make sure they are properly configured for the motor on the new device.

| Button or Dialog Box | Description |
|----------------------|---|
| Save To File | Copies all the device parameters and creates a file. |
| Load From Device | Sets all the device parameters within the device to the values contained within the file you select. |
| Save to Device | Saves all device parameters into the nonvolatile memory of the device. Each time the device powers on, it will start with these saved parameters. |
| Factory Defaults | Returns all the device parameters back to their default values. Any changes you have made are lost. |

14.3.1 Loading an Incompatible File

If the parameter file from a different model type is loaded to an AKD PDMM, a warning will display notifying the user, but the load can still be performed.

15 Faults and Warnings

| | | |
|-------------|---|------------|
| 15.1 | Fault and Warning Messages | 177 |
| 15.2 | AKD PDMM Errors and Alarms | 193 |
| 15.3 | Additional error and alarm messages AKD PDMM | 194 |
| 15.4 | Clearing Faults | 199 |
| 15.5 | Parameter and Command Error Messages | 199 |
| 15.6 | Unknown Fault | 203 |

15.1 Fault and Warning Messages

When a fault occurs, the drive fault relay is opened, the output stage is switched off (motor loses all torque), or the load is dynamically braked. The specific drive behavior depends on the type of fault. The LED display on the front panel of the drive shows the number of the fault that occurred. If a warning is issued prior to the fault, the warning is shown on the LED and has the same number as the associated fault. Warnings do not trip the power stage of the drive or fault relay output.

The left side of the LED displays F for a fault or n for a warning. The right side displays the fault or warning number as follows: 1-0-1-[break]. The highest priority fault is displayed on the LED. Multiple faults may be present when a fault condition is occurring. Check the AKD PDMM EWV¹ Fault Screen or read the status of DRV.FAULTS through the controller or HMI for the entire list of faults.

| Fault | Message/Warning | Cause | Remedy |
|-------|---|---|---|
| .. | | 1. 24V Control Power input voltage dip. or 2. Auxillary encoder 5V (X9-9) shorted. | 1. Insure adequate 24V supply current capacity for the system. or 2. Check and fix X9 wiring. |
| F0 | | Reserved. | N/A |
| F101 | Firmware incompatible. | Installed firmware is not compatible with the drive hardware. | Load compatible firmware into the drive. |
| n101 | The FPGA is a lab FPGA. | The FPGA is a lab version FPGA. | Load the released FPGA version that is compatible with the operational firmware. |
| F102 | Resident Firmware failed. | Software failure detected. | Restart drive. If issue persists, contact technical support. |
| n102 | Operational FPGA is not a default FPGA. | The FPGA minor version is larger than the operational firmware default FPGA minor version | Load the released FPGA version that is compatible with the operational firmware. |
| F103 | Resident FPGA failed. | Software failure detected. Load resident FPGA failure occurred (several cases according to flowchart, including incompatible image to FPGA type and fieldbus type). | Restart drive. If issue persists, contact technical support. |
| F104 | Operational FPGA failed. | Software failure detected. Load operational FPGA failure occurred (several cases according to flowchart). | Restart drive. If issue persists, contact technical support. |
| F105 | NV Memory Stamp. | Non-volatile memory stamp is corrupted or invalid. | Reset the drive to default memory values using Parameter Load in EWV. |

¹Embedded Workbench Views

| Fault | Message/Warning | Cause | Remedy |
|--------------|---|---|---|
| F106 | NV Memory Data (might occur when downloading firmware). | Non-volatile memory data is corrupted or invalid. When this fault occurs after a firmware download, it is not an indication of a problem (clear the fault and perform a “save” to the drive). | Reset the drive to default memory values using Parameter Load in EWW. |
| n107 | Positive switch. | Positive position limit is exceeded. | Move the load away from the limits. |
| n108 | Negative switch. | Negative position limit is exceeded. | Move the load away from the limits. |
| F121 | Homing Error. | Drive did not finish homing sequence. | Check homing sensor, homing mode, and homing configuration. |
| F123 n123 | Invalid motion task. | Invalid motion task. | Check motion task settings and parameters to make sure that the values entered will produce a valid motion task. |
| F125 n125 | Fieldbus Sync frames lost. | The fieldbus lost synchronization. | Check fieldbus connection (X5 and X6 if you are using EtherCAT; X12 and X13 if you are using CANopen) or the settings of your EtherCAT or CANopen master. |
| F126 n126 | Bode plot too much movement. | Too much movement was created during a Bode plot. Motor is unstable and is not following drive instructions. | Check that the system is closed loop stable. Refer to the system tuning guide. |
| F127 | Incomplete Emergency Procedure. | Incomplete emergency stop procedure (problem with the emergency stop motion task). | Disconnect power from drive and check emergency stop procedure. |
| F128 | Mpoles / Fpoles not integer. | Ratio of motor poles to feedback poles must be a whole number. | Change to a compatible feedback device. |
| F129 | Fieldbus Heartbeat lost. | Heartbeat lost. | Check CANopen cabling. Reduce bus load or increase the heartbeat update time. |
| F130 | Secondary feedback supply over current. | 5V power supply was shorted out on X9. | Check X9 connection. |
| F131 | Secondary feedback A/B line break. | Problem in secondary feedback detected. | Check secondary feedback (X9 connection). |
| F132 | Secondary feedback Z line break. | Problem in secondary feedback detected. | Check secondary feedback (X9 connection). |

| Fault | Message/Warning | Cause | Remedy |
|--------------|--|--|---|
| F133 | Fault number changed to F138. See F138 for details. | | |
| F134 | Secondary feedback illegal state. | Feedback signals were detected in an illegal combination. | Check X9 connection. |
| F135 n135 | Fault on FB2 is preventing remote commutation for FB1. | Attempt to issue motion task before the axis is homed. Axis must be homed before motion task can start. | Change opmode or home axis. |
| F136 | Firmware and FPGA versions are not compatible. | The FPGA version does not match the firmware FPGA version constants. | Load the FPGA version that is compatible with the firmware. |
| n137 | Homing and feedback mismatch | The configured homing mode is not supported by the motor feedback type used. | Change homing mode. |
| F138 | Instability during autotune. | Drive current (IL.CMD) or velocity feedback (VL.FB) exceeds allowable limit (BODE.IFLIMIT or BODE.VFLIMIT). This fault only occurs in BODE.MODE 5. This fault often occurs when complex mechanics, belts, and compliant loads are present. | Change BODE.MODE if appropriate. If BODE.MODE 5 is appropriate and the fault occurs at the end of an Autotuning, then the motor is not robustly stable. You can manually adjust Autotuner settings. Manual tuning may be required to make the motor stable. |
| F139 | Target position was overshoot due to invalid motion task activation. | The drive cannot decelerate from its current speed to reach the end point of the second motion task without moving past it. Increase the deceleration rate in the move or trigger the move earlier. | Change Motion task profile and clear fault with DRV.CLRFAULTS. Or change the value of FAULT139.ACTION = 1 to ignore this condition. |
| n140 | VBUS.HALFVOLT has changed. Save the parameters and reboot the drive. | The user has changed the numerical value of VBUS.HALFVOLT. This change only takes effect after a DRV.NVSAVE command and after rebooting the AKD PDMM. | Save the parameters in the non-volatile memory via a DRV.NVSAVE command and turn off/on the 24[V] power supply in order to reboot the drive or restore the original setting of VBUS.HALFVOLT. |

| Fault | Message/Warning | Cause | Remedy |
|-------|--|---|--|
| n151 | Not enough distance to move; motion exception. | <p>For trapezoidal and customer table motion tasks: The target velocity specified in the motion task cannot be reached via using the selected acceleration and deceleration since the distance to travel is not sufficient.</p> <p>For a 1:1 profile: The selected acceleration and deceleration will be extended since there is too much distance to travel and the motion task would exceed its maximum allowed velocity.</p> | Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. Check motion task settings and parameters to make sure that the values entered will produce a valid motion task. |
| n152 | Not enough distance to move; following motion exception. | A new motion task activated, when one motion task is already active and the target position specified in the motion task parameters cannot be reached with specified target velocity, acceleration and deceleration parameters. The motion task will directly decelerate to into the target position or ramps down to velocity 0 and start another move to reach target position of the next motion task. | Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. Check motion task settings and parameters to make sure that the values entered will produce a valid motion task. |
| n153 | Velocity limit violation, exceeding max limit. | A new target velocity calculated internally due to an exception, and is being limited due to user velocity limit. | Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. Check motion task target velocity settings and parameters to make sure that the values entered will not exceed the VL.LIMITP and VL.LIMITN setting. |
| n154 | Following motion failed; check motion parameters. | Activation of the following motion task failed due to incompatible parameters, or motion task does not exist. | Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. Check following motion task settings and parameters to make sure that the values entered will produce a valid motion task. |

| Fault | Message/Warning | Cause | Remedy |
|-------|--|---|---|
| n156 | Target position crossed due to stop command. | The motion task crosses the target position after triggering a DRV.STOP command. This situation can happen when processing a change-on-the-fly motion task and triggering a DRV.STOP command close to the target position of the currently running motion task. | Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. |
| n157 | Homing index pulse not found. | A homing mode with index detection is activated, and index pulse is not detected while moving across the range determined by the hardware limit switches. | Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. |
| n158 | Homing reference switch not found. | A homing mode with reference switch detection is activated and the reference switch is not detected while moving across the range determined by the hardware limit switches. | Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. |
| n159 | Failed to set motion task parameters | Invalid motion task parameters assignment. This warning can appear upon an MT.SET command. | Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. Check motion task settings and parameters. |
| n160 | Motion task activation failed. | Activation of the motion task failed due to incompatible parameters, or motion task does not exist. This warning can appear upon an MT.MOVE command. | Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. Check motion task settings and parameters to make sure that the values entered will produce a valid motion task. |
| n161 | Homing procedure failed. | Homing error observed during the operation of homing procedure. | Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. |
| n163 | MT.NUM exceeds limit. | This warning appears with n160. This warning is triggered when you try to trigger a motion task > 128 (such as MT.MOVE 130). | Trigger only motion tasks between 0 and 128. Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. |

| Fault | Message/Warning | Cause | Remedy |
|-------------|--|--|---|
| n164 | Motion task is not initialized. | This warning appears with n160. This warning is triggered when you try to trigger a non-initialized motion task. | Initialize the motion task first before starting the task. Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. |
| n165 | Motion task target position is out. | This warning appears with n160. This warning is triggered when you try to trigger a motion task with an absolute target position outside of the selected modulo range (see also MT.CNTL). | Move the absolute target position of the motion task within the modulo range. Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. |
| n168 | Invalid bit combination in the motion task control word. | This warning appears with n160. This warning is triggered when you try to trigger a motion task with an invalid bit combination in the motion task control word (see also MT.CNTL). | Correct the MT.CNTL setting for the specific motion task. Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. |
| n169 | 1:1 profile cannot be triggered on the fly. | This warning appears with n160. This warning is triggered when you try to trigger a 1:1 profile table motion task while another motion task is currently running. | 1:1 profile table motion tasks should be started from velocity 0. Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. |
| n170 | Customer profile table is not initialized. | This warning appears with n160. This warning is triggered when you try to trigger a motion task that uses a customer profile table for generating the velocity profile and when the selected profile table is empty (see MT.CNTL and MT.TNUM). | Change the MT.TNUM parameter for this specific motion task in order to use an initialized profile table. Activation of any new motion or using of DRV.CLRFAULTS will clear the warning. |
| F201 | Internal RAM failed. | Hardware failure detected. | Restart drive. If issue persists, contact technical support. |
| F202 | External RAM failed. | Hardware failure detected. | Restart drive. If issue persists, contact technical support. |
| F203 | Code Integrity failed. | Software failure detected. FPGA register access failure occurred. | Restart drive. If issue persists, contact technical support. |
| F204 - F232 | EEPROM failure detected | EEPROM failure detected | Restart drive. If issue persists, exchange drive. |

| Fault | Message/Warning | Cause | Remedy |
|----------------------------|---|--|---|
| F234 - F237 n234 - n237 | Temperature sensor high. | High temperature limit reached. | Check cabinet ventilation system. |
| F240 - F243 n240 - n243 | Temperature sensor low. | Low temperature limit reached. | Check cabinet ventilation system. |
| F245 | External Fault. | This fault is user generated and is caused by user settings. | Users can configure a digital input to trigger this fault (DINx.MODE = 10). The fault occurs according to this input setting. Clear the input to clear the fault. |
| F247 | Vbus read is out of thresholds. | Hardware problem in bus measurement. | Troubleshoot and repair hardware problem. |
| F248 | Option board EEPROM corrupted. | EEPROM failure detected. | Restart drive. If issue persists, exchange drive. |
| F249 | Option board downstream checksum. | Communications with the I/O on the option board failed. | DRV.CLRFAULTS. If issue persists If issue persists, contact technical support. |
| F250 | Option board upstream checksum. | Communications with the I/O on the option board failed. | DRV.CLRFAULTS. If issue persists If issue persists, contact technical support. |
| F251 | Option board watchdog. | Communications with the I/O on the option board failed. | DRV.CLRFAULTS. If issue persists If issue persists, contact technical support. |
| F252 | Firmware and option board FPGA types are not compatible. | The option board FPGA is not compatible with this hardware. | Download the correct firmware file for this drive. |
| F253 | Firmware and option board FPGA versions are not compatible. | The version of the option board FPGA is not compatible with this firmware. | Download the correct firmware file for this drive. |
| F256 | Analog Input over voltage | The analog input value is above the AIN.OVFTHRESH level. | Check the analog input signal or change the AIN.OVFTHRESH value. |
| F257 | Analog Input under voltage | The analog input value is below the AIN.UVFTHRESH level. | Check the analog input signal or change the AIN.UVFTHRESH value. |
| n256 | Warning: Analog Input over voltage | The analog input value is above the AIN.OVWTHRESH level. | Check the analog input signal or change the AIN.OVWTHRESH value. |
| n257 | Warning: Analog Input under voltage | The analog input value is below the AIN.UVWTHRESH level. | Check the analog input signal or change the AIN.UVWTHRESH value. |
| F301 n301 | Motor overheated. | Motor overheated. | Check ambient temperature. Check motor mounting heat sink capability |
| F302 | Over speed. | Motor exceeded VL.THRESH value. | Increase VL.THRESH or lower velocity command. |

| Fault | Message/Warning | Cause | Remedy |
|--------------|---|---|--|
| F303 | Run away. | Motor did not follow command values. | Current command to the motor is too high for too long. Reduce servo gains, or reduce command trajectory aggressiveness. |
| F304 n304 | Motor Foldback. | Maximum motor power has been exceeded; the power has been limited to protect the motor | Motion is requiring too much power. Change move profile to reduce load on motor. Check for load jamming or sticking. Check that current limits are set correctly. |
| F305 | Brake open circuit. | Motor brake open circuit. Fault threshold is 200 mA. | Check cabling and general functionality. For special low current brake applications, the F305 fault can be bypassed using the setting <code>motor.brake = 100</code> . |
| F306 | Brake short circuit. | Motor brake short circuit. | Check cabling and general functionality. Check that <code>MOTOR.TBRAKERLS</code> (→ p. 623) and <code>MOTOR.TBRAKEAPP</code> (→ p. 622) are configured. |
| F307 | Brake applied during Enable-State. | Motor brake closed unexpectedly. | Check cabling and general functionality. |
| F308 | Voltage exceeds motors rating. | Drive bus voltage exceeds the motor's defined voltage rating. | Make sure that the motor fits the driving rating. |
| n309 | Motor I2t load. reduce load | Motor I2t load (IL.MI2T) has exceeded the warning threshold <code>IL.MI2TWTHRESH</code> . This warning can only be generated in the case that the motor protection mode <code>IL.MIMODE</code> has been set to 1. | Reduce the load of the drive by adjusting lower acceleration / deceleration ramps. |
| F312 | Brake released when it should be applied. | Brake disengaged unexpectedly. | Check cabling and general functionality. |
| F401 | Failed to set feedback type. | Feedback is not connected or wrong feedback type selected | Check primary feedback (X10 connection). |
| F402 | Analog signal amplitude fault. | Analog signal amplitude is too low. Analog fault (resolver signal amplitude or sin/cos amplitude) | Check primary feedback (X10 connection), resolver and sine/cos encoder only. |
| F403 | EnDat communication fault. | General communication problem with feedback. | Check primary feedback (X10 connection), EnDat only. If feedback is linear, check calibration. |

| Fault | Message/Warning | Cause | Remedy |
|--------------|--|---|---|
| F404 | Illegal Hall state (111, 000). | Hall sensor returns invalid Hall state (111, 000); either all Hall sensors are on or off. Legal Hall states are 001, 011, 010, 110, 100, and 101. This fault can be caused by a broken connection in any one of the Hall signals. | Check the feedback wiring; check all feedback connectors to ensure all pins are positioned correctly. |
| F405 | BiSS watchdog fault. | Bad communication with the feedback device. | Check primary feedback (X10 connection), Biss only. |
| F406 | BiSS multi cycle fault. | | |
| F407 | BiSS sensor fault. | | |
| F408 - F416 | SFD feedback fault. | Bad communication with the SFD device. | Check primary feedback (X10 connection). If fault persists, internal feedback failure. Return to manufacturer for repair. |
| F417 | Broken wire in primary feedback. | In primary feedback, a broken wire was detected (incremental encoder signal amplitude). | Check feedback cable continuity. |
| F418 | Primary feedback power supply. | Power supply fault for primary feedback. | Check primary feedback (X10 connection). |
| F419 | Encoder init procedure failed. | Phase find procedure did not complete successfully. | Check encoder wiring, reduce/balance motor load prior to phase finding. |
| F420 | FB3 Endat communication fault. | A communication error was detected with the EnDat 2.2 device connected to the X9 connector. | Check pinout and FB3 configuration and reconnect feedback. If problems persist, contact customer support. |
| F421 | SFD resolver sensor fault. | Sensor or sensor wiring failure inside motor. | Try resetting the fault. If it reappears return motor for repair. |
| F423 | NV Failure – Extended Multiturn. | The position saved in memory is corrupted. | Home axis or disable extended multiturn. If the fault persists, send the drive for repair. |
| F438 n438 | Deviation from predicted trajectory fault. | Motor did not follow command values. Motor exceeded maximum allowed position following error (numeric). | Check for increased load, jamming or sticking. Is position error set too low? |
| F439 n439 | Following error magnitude fault. | Motor did not follow command values. Motor exceeded maximum allowed position following error (user). | Check feedback commutation setup and tuning parameters. |

| Fault | Message/Warning | Cause | Remedy |
|--------------|--|--|---|
| F450 | Following error presentation. | Motor did not follow command values. Motor exceeded maximum allowed position following error (presentation). | Check feedback commutation setup and tuning parameters. |
| F451 n451 | Feedback battery fault. | The external battery voltage is too low. The F451 fault is generated if the AKD PDMM is not powered. The n451 warning is generated if the AKD PDMM is powered. This fault can be inhibited with FAULT451.ACTION. | Check or replace the external battery. |
| F452 | Extended Multiturn not supported with this feedback. | Non-multiturn feedback is connected while FB1.PMTSAVEEN is active. | Connect multiturn feedback to the drive or disable extended multiturn. |
| F453 | Tamagawa communication fault (timeout). | Bad communication with the feedback device. Cabling or shielding fault, or internal feedback failure. | Check the cabling to the drive and if the problem persists then return the feedback to the manufacturer for repair. |
| F454 | Tamagawa communication fault (transfer incomplete). | | |
| F456 | Tamagawa communication fault (CRC). | | |
| F457 | Tamagawa communication fault (start timeout). | | |
| F458 | Tamagawa communication fault (UART Overrun). | | |
| F459 | Tamagawa communication fault (UART Framing). | | |
| F460 | Tamagawa encoder fault (over speed). | This fault is generated when the shaft is rotated above a maximum speed that can be maintained while the external battery is powered and the drive is powered off. | Reset the fault on the drive with DRV.CLRFAULTS. |
| F461 | Tamagawa encoder fault (counting Error). | When the feedback is powered on the position (within one revolution) was incorrect because of a problem with the feedback device. | Reset the fault on the drive with DRV.CLRFAULTS, if the problem persists then clean the feedback code plate. |
| F462 | Tamagawa encoder fault (counting overflow). | Multi-turn counter has overflowed. | Reset the fault on the drive with DRV.CLRFAULTS. |

| Fault | Message/Warning | Cause | Remedy |
|-------|--|--|---|
| F463 | Feedback overheat fault. | The temperature of the encoder substrate exceeds overheating detection temperature during main power-on. | Reset the fault on the drive with DRV.CLRFAULTS after temperature of encoder is lowered. |
| F464 | Tamagawa encoder fault (multi-turn error). | Any bit-jump occurs in the multi-turn signal during main power-on. | Return to the origin. Reset the fault on the drive with DRV.CLRFAULTS. |
| F465 | Excessive shock detected by feedback device. | 1. Excessive shock from an impact or vibration has caused an error in the feedback device. or 2. Internal error in the feedback device mechanism, resulted in bad position data. | 1. Reduce external impacts to the motor housing and shaft. Tune the control loops filters. Reduce gains, particularly feed forward gains. Reduce maximum commanded acceleration. or 2. Replace the feedback device. |
| F467 | Feedback fault on feedback 1 (See FB1.FAULTS for details). | The feedback device malfunctioned. | Check FB1.FAULTS for detailed fault information. If using a BiSS feedback device, fault 467 indicates a communications fault with the BiSS feedback device. No additional information is available via FB1.FAULTS for this fault when using a BiSS feedback device. |
| F469 | FB1.ENCRES is not power of two, Remote Commutation not possible. | Feedback Type 43 requires that the feedback resolution be a power of two. Feedback Type 43 is not supported for all feedback resolutions. | If the remote feedback device's resolution is a power of two, enter that value in FB1.ENCRES. Otherwise, choose a feedback device with a supported resolution (power of two) and enter that value in FB1.ENCRES. |
| F470 | Feedback fault on feedback 3. | Feedback is not connected or general communication problem. | Check tertiary feedback (X9 connection) |
| F473 | Wake and Shake - Insufficient movement | There was less movement than defined by WS.DISTMIN. | Increase WS.IMAX and/or WS.T. Or try using WS.MODE 1 or 2. |
| F475 | Wake and Shake - Excess movement | WS.DISTMAX has been exceeded in WS.MODE 0. Or more than 360 degrees was traveled in WS.MODE 2. | Increase WS.DISTMAX value or reduce WS.IMAX or WS.T. Wake and Shake is not supported for vertical/overhung loads. |

| Fault | Message/Warning | Cause | Remedy |
|--------------|--|--|--|
| F476 | Wake and Shake - Fine-Coarse delta too large. | The angle difference between the coarse and fine calculation was larger than 72 deg. | Modify WS.IMAX or WS.T and try again. |
| F478 n478 | Wake and Shake - Over speed | WS.VTHRESH was exceeded. | Increase WS.VTHRESH value or reduce WS.IMAX or WS.T. |
| F479 n479 | Wake and Shake - Loop angle delta too large. | The angle between complete loops was larger than 72 deg. | Modify WS.IMAX or WS.T and try again. |
| F480 | Field bus command velocity too high. | Fieldbus command velocity exceeds VL.LIMITP. | Lower fieldbus command trajectory, or increase the value of VL.LIMITP. |
| F481 | Field bus command velocity too low. | Fieldbus command velocity exceeds VL.LIMITN. | Increase fieldbus command trajectory, or decrease the value of VL.LIMITN. |
| F482 | Wake and Shake - Commutation not initialized. | The motor requires the commutation initialization (there are no encoder commutation tracks, Hall sensors, etc.) and no successful Wake and Shake sequence has been performed | Clear any faults, activate the Wake and Shake procedure (WS.ARM) and enable the drive. |
| F483 | Wake and Shake - Motor U phase missing. | No current was detected in the motor's U phase during Wake and Shake initialization (Mode 0 only). | Check the motor connections and WS.IMAX (very low current may produce this error). |
| F484 | Wake and Shake - Motor V phase missing. | No current was detected in the motor's V phase during Wake and Shake initialization (Mode 0 only). | Check the motor connections and WS.IMAX (very low current may produce this error). |
| F485 | Wake and Shake - Motor W phase missing. | No current was detected in the motor's W phase during Wake and Shake initialization (Mode 0 only). | Check the motor connections and WS.IMAX (very low current may produce this error). |
| F486 | Input change rate exceeds maximum speed of emulated encoder. | Motor velocity exceeds the maximum speed the emulated encoder output can generate. | Reduce value of DRV.EMUEPULSEWIDTH. |
| F487 | Wake and Shake - Validating Positive Movement Failed. | After applying a positive current, motor moved in the wrong direction. | Check motor phase wiring and motor encoder wiring is correct. |
| F489 | Wake and Shake - Validating Negative Movement Failed. | After applying a negative current, motor moved in the wrong direction. | Check motor phase wiring and motor encoder wiring is correct. |

| Fault | Message/Warning | Cause | Remedy |
|--------------|---|--|--|
| F490 | Wake and Shake - Validating Comm. angle timed out. | During one of the W&S validation stages, the drive stopped responding to commands. | Contact customer support. |
| F491 | Wake and Shake - Validating Comm. angle moved too far - Bad Comm Angle. | After applying a current, the motor moved too far (>15 electrical degrees). | This indicates a poor motor phase angle was found by Wake and Shake. Revise Wake and Shake parameters, and re-run Wake and Shake. |
| F492 | Wake and Shake - Validating Comm. angle required more than MOTOR.ICONT | A current larger than MOTOR.ICONT was used to excite the motor. | This indicates one of the following: <ul style="list-style-type: none"> Phase angle is incorrect due to a bad wake and shake. Motor has very high friction requiring high current to break free. Motor power cable is disconnected or improperly wired. |
| F493 | Invalid commutation detected - motor accelerating in the wrong direction. Motor phase may be incorrect. | The velocity of the motor exceeded WS.CHECKV and the sign of the current was not equal to the sign of motor acceleration or the sign of motor velocity for a period of time larger than WS.CHECKT. | 1. Check motor phase wiring 2. Re-configure wake and shake (if Mode 0 or 1 is used) 3. Re-run wake and shake to determine correct commutation angle |
| F501 n501 | Bus over voltage. | Bus voltage too high. Usually, this problem is load related. | Reduce load or change motion profile. Check system regen capacity; add capacity if needed. Check mains voltage. |
| F502 | Bus under voltage. Warning issued prior to fault. | Bus voltage below threshold value. | Check mains voltage. |
| F503 n503 | Bus capacitor overloaded. | Single phase AC input on a drive only rated for three-phase input or excessive single-phase power load. | Check mains voltage. |
| F504 - F518 | Internal supply voltage fault | Internal supply voltage fault detected | Check wiring for electromagnetic compatibility (EMC). If issue persists exchange drive. |
| F519 | Regen short circuit. | Regen resistor short circuit. | Regen IGBT short circuit. Contact technical support. |

| Fault | Message/Warning | Cause | Remedy |
|--------------|-----------------------------------|---|---|
| F521 n521 | Regen over power. | Too much power stored in regen resistor. | Either get larger regen resistor or use DC bus sharing to dissipate power. |
| F523 | Bus over voltage FPGA. | Bus over voltage hard fault. | Check mains voltage and check system brake capacity. |
| F524 n524 | Drive Foldback. | Maximum drive power has been exceeded. The power has been limited to protect the drive. | Motion requires too much power. Change profile to reduce load . |
| F525 | Output over current. | Current exceeds drive peak. | Check for short or feedback faults. |
| F526 | Current sensor short circuit. | 1. Current sensor short circuit. 2. Motor cable plugged in upside down | 1. Restart drive. If issue persists, contact technical support. 2. Plug in cable right-side up. |
| F527 | Iu current AD converter stuck. | Hardware failure detected. | Restart drive. If issue persists, contact technical support. |
| F528 | Iv current AD converter stuck. | Hardware failure detected. | Restart drive. If issue persists, contact technical support. |
| F529 | Iu current offset limit exceeded. | Hardware failure detected. | Restart drive. If issue persists, contact technical support. |
| F530 | Iv current offset limit exceeded. | Hardware failure detected. | Restart drive. If issue persists, contact technical support. |
| F531 | Power stage fault. | Hardware failure detected. | Restart drive. If issue persists, replace drive. |

| Fault | Message/Warning | Cause | Remedy |
|-------|---|---|--|
| F532 | Drive motor parameters setup incomplete. | Before a motor can be enabled, you must configure a minimum set of parameters. These parameters have not been configured. | Issue the command <code>DRV.SETUPREQLIST</code> to display the list of the parameters that you must configure. Configure these parameters either manually or automatically. You can manually configure these parameters in three ways: (1) set each parameter individually; (2) use the setup wizard to select the motor; or (3) select the motor type from the motor data base in the Motor window (<code>MOTOR.AUTOSSET</code> must be set to 0 (off)). If you use the Motor window, you must first select the feedback type. If the motor has Biss Analog, Endat, or SFD feedback (feedback with memory), then these parameters are set automatically when <code>MOTOR.AUTOSSET</code> is set to 1 (on). |
| F534 | Failed to read motor parameters from feedback device. | Motor either does not have motor feedback memory, or the motor feedback memory is not programmed properly so the parameters cannot be read. | Try to read parameters again by clicking the Disable and Clear Faults button, or by issuing the <code>DRV.CLRFAULTS</code> command. If this attempt is not successful, then set <code>MOTOR.AUTOSSET</code> to 0 (off) and program the parameters using the setup wizard or manually set up the parameters. If the motor has motor memory (Biss Analog, Endat, and SFD motors have motor memory), return the motor to have the memory programmed. |
| F535 | Power-board over-temperature fault. | The power-board temperature sensor indicates more than 85 °C. | Reduce the load of the drive or ensure better cooling. |
| F536 | Standby power supply fault. | Standby voltage circuit overloaded. | Check total 24 V power load of AKD PDMM-N in the strings (sum of drive and motor brake supply). The AKD PDMM-C must be power-cycled to recover from this fault. |

| Fault | Message/Warning | Cause | Remedy |
|-------|---|---|---|
| F537 | Precharge fault. | Precharge process could not be performed in a reasonable amount of time. | Look for a shortcut in the string (cable) or exchange hardware |
| F541 | AC input phase 1 missing. | Mains supply failure detected, phase L1 missed. | Check power connector and power source. |
| F542 | AC input phase 2 missing. | Mains supply failure detected, phase L2 missed. | Check power connector and power source. |
| F543 | AC input phase 3 missing. | Mains supply failure detected, phase L3 missed. | Check power connector and power source. |
| F545 | String current over peak limit | String current is higher than AKD PDMM-C Rated Peak Current. | Lower AKD PDMM-N current limits to prevent over-drawing AKD PDMM-C current |
| F546 | String current over continuous limit | String current is higher than AKD PDMM-C Rated Continuous Current (I^2T). | Lower AKD PDMM-N current limits to prevent over-drawing AKD PDMM-C current |
| F560 | Regen near capacity, could not prevent over voltage. | An F501 Bus Over Voltage has occurred while the Regen Resistor was at or above 75% of its dissipation capacity. | Increase the size of regen resistor to be able to dissipate more power. |
| F561 | More than 8 AKD PDMM-Ns connected at string2 | Too many drives on string 2. | Reduce NSDs on String 2 to 8 or less |
| F562 | More than 8 AKD PDMM-Ns connected at string1 | Too many drives on string 1. | Reduce NSDs on String 1 to 8 or less |
| F564 | Number of connected nodes on string 1 has decreased. | Drive has been removed from string 1. | Investigate AKD PDMM-N Ethercat communication, determine where network connection failed. |
| F565 | Number of connected nodes on string 2 has decreased. | Drive has been removed from string 2. | Investigate AKD PDMM-N Ethercat communication, determine where network connection failed. |
| F570 | Phase loss. | Phase loss detected. | Check mains power voltage. Fault action configurable by FAULT570.ACTION (→ p. 453). |
| n582 | Velocity has been limited commutation speed to less than 600Hz to meet ECCN 3A225 requirements. | Motor Velocity has exceeded the allowed commutation speed (599Hz). | Refer to ECCN 3A225 Limitations for Induction Motors. |
| n601 | Modbus data rate is too high. | Modbus controller data rate is too high. | Reduce data rate. |

| Fault | Message/Warning | Cause | Remedy |
|--------------|--|--|--|
| F602 | Safe torque off. | Safe torque off function has been triggered. | Reapply supply voltage to STO if safe to do so. |
| n603 | OPMODE incompatible with CMDSOURCE | This warning is generated when the drive is enabled and the gearing command source is selected at the same time as torque or velocity op-mode. | Select a different DRV.OPMODE and DRV.CMDSOURCE combination. |
| n604 | EMUEMODE incompatible with DRV.HANDWHEELSRC. | Emulated encode mode is incompatible with the selected handwheel source. | Select a compatible emulated encode mode or change handwheel source. |
| F621 | Control Board CRC fault. | Communications with the Power Board Failed | DRV.CLRFAULTS. If issue persists, contact technical support. |
| F623 | Power Board CRC fault. | | |
| F624 | Power Board Watchdog fault. | | |
| F625 | Power Board Communication fault. | | |
| F626 | Power Board FPGA not configured. | | |
| F627 | Control Board Watchdog fault. | | |
| F628 | AKD PDMM-C Front door packet not received on String 1. | A data packet has not been received by the AKD PDMM-N or AKD PDMM-C (String 1) | Check cables and clear the fault. |
| F629 | AKD PDMM-C Front door packet not received on String 2. | A data packet has not been received by the AKD PDMM-C (String 2) | Check cables and clear the fault. |
| F630 | FPGA cyclic read fault. | FPGA-to-firmware data access error. | DRV.CLRFAULTS. If issue persists, contact technical support. |
| F701 | Fieldbus Runtime. | Runtime communication fault. | Check fieldbus connections (X11), settings, and control unit. |
| F702 n702 | Fieldbus Communication lost. | All fieldbus communication was lost. | Check fieldbus connections (X11), settings, and control unit. |
| F703 | Emergency timeout occurred. | Motor did not stop in the timeout defined. | Change timeout value, change stop parameters, improve tuning. |


15.2 AKD PDMM Errors and Alarms

When on or more Errors occur, the highest priority Error is displayed on the 7-segment LED of the controller. Error messages have priority over Alarm messages. When an Error or Alarm occurs, always check the controller log messages. The log messages will provide more details about the failure and the history of events

leading up to the failure. From the log messages, you can determine the specifics about the cause of the failure to correct the underlying problem.

15.3 Additional error and alarm messages AKD PDMM

Faults/Errors and Warnings/Alerts are displayed in the 7 segment displays of the drive:

| AKD PDMM two digits | AKD PDMM-M two + one digits |
|---|--|
|  | |
| <p>The two digits LED display indicates the drive messages. Drive fault messages are coded with "F", warnings are coded with "n" and are described in "Fault and Warning Messages" (=> p. 1)</p> | <p>The one digit LED indicates the PDMM messages of the AKD PDMM. Error messages are coded with "E", alarms are coded with "A" and are described in the chapter below.</p> |

To simplify handling, the error and alarm handling process is consistent, so you can always apply the same recovery steps. When an error or alarm occurs, it is displayed to the one digit display, you can identify the error in the table below and follow the recommendations to fix the problem, clear the display, and resume machine operation.

Active errors and alarms can be cleared with the controller command **ClearCtrlErrors**, (Note: non-clearable errors will remain).

15.3.1 Additional error and alarm messages AKD PDMM-M

Faults/Errors and Warnings/Alerts are displayed in the 7 segment displays of the drive. When an error or alarm occurs, it is displayed to the one digit display, you can identify the error in the table below and follow the recommendations to fix the problem, clear the display, and resume machine operation. Active errors and alarms can be cleared with the controller command **ClearCtrlErrors**, non-clearable errors will remain.

More information about error messages, causes, remedy and clearing errors can be found in the [EWV¹](#) online help and in [KDN](#).

15.3.2 Errors

When an Error or Alarm occurs, always check the controller log messages. The log messages will provide more details about the failure and the history of events leading up to the failure. From the log messages, you can determine the specifics about the cause of the failure to correct the underlying problem.

Eliminate errors and faults in compliance with work safety rules. Troubleshooting only by qualified and trained staff.

More information about error and alarm messages, causes, remedy and clearing errors can be found in the [EWV](#) online help and in [KDN](#).

¹Embedded Workbench Views

| Error | Description | Cause | Remedy |
|-------|--|---|--|
| E01 | Critical temperature exceeded. Operation is stopped, CPU will be put to sleep. | CPU temperature exceeded safe operating temperature limit. | Power-off. Check airflow and operating environment are within hardware specifications. Allow unit to cool before power-on. |
| E02 | Out of memory. KAS runtime is stopping. | Memory leak, memory corrupted, or hardware memory failure. | Power-off/on. If problem is recurrent, check release notes for firmware updates or return hardware for repair. |
| E03 | Fan failure. | CPU cooling fan was not able to operate properly. | Check temperature and monitor for High temp alarm (see A01). Return hardware for fan replacement. |
| E10 | Firmware is corrupted. | Flash memory corrupted during firmware download or flash hardware failure. | Re-download firmware or boot into recovery mode, download. If the problem is recurrent, power-off/on. Check release notes for firmware updates. firmware, and power-off/on. If problem persists, return hardware for repair. |
| E11 | Flash is corrupted, no filesystem is available. | At startup the filesystem could not be mounted on the flash. | Reset to factory defaults. If problem persists, return hardware for repair. |
| E12 | Not enough flash memory available. | Flash memory is full, unable to write to flash. | Clean-up the flash memory by removing log files, application programs, recipes, or other data files. |
| E13 | Out of NVRAM space for retained variables. | NVRAM is full. | Change application to reduce the amount of retained variables. |
| E14 | Reset to Factory Defaults failed. | Flash memory could not be formatted during a Reset to Factory Defaults procedure. | Try reset to factory defaults again from power-on. If problem persists, return hardware for repair. |
| E15 | Cannot read/write files from/to a SD card. | SD card is not plugged-in or the filesystem is corrupt and cannot be mounted. | Insert a valid SD card or re-format the SD card using Settings->SD card->Format button. |
| E16 | Not enough space available on the SD card. | SD card is full, unable to write to the SD card. | Clean-up SD card space by deleting files or re-format the card using Settings->SD card->Format button. |
| E20 | Runtime plug-in, process, thread or application failed to start. | KAS runtime or application code failed to auto-start at boot. | Power-off/on. Reset to factory defaults. If problem is recurrent, check release notes for firmware updates or download firmware. |
| E21 | Runtime plug-in, process, or thread failed to respond during operation. | KAS runtime code failed during normal operation. | Power-off/on. If problem is recurrent, check release notes for firmware updates. |

| Error | Description | Cause | Remedy |
|-------|---|--|--|
| E22 | Fatal error in PLC program, application stopped. | Virtual machine failed to execute an instruction. | Re-compile application, download, and re-start. |
| E23 | CPU is over-loaded. | Either the motion engine cycle did not complete or the PLC program did not complete within the timeout period due to excessive CPU loading. | Stop the application or power-off/on. Reduce the sample rate, simplify the application, or reduce the application cycles and re-start the application. |
| E24 | PLC application cannot be started | <ul style="list-style-type: none"> • Maintenance operation is in progress. • Controller is in online config mode. • AKD PDMM Restore failed. • The IDE version of the compiled PLC code and controller runtime version do not match. | <ol style="list-style-type: none"> 1. Check controller web-server home page for any maintenance operation in-progress. Wait for the operation to finish. 2. Connect to the controller with the IDE and disable online config mode. 3. Check EtherCAT network topology by using the Scan network button in the web-server's Restore tab. Correct the physical topology and re-execute an AKD PDMMrestore. 4. IDE version (only major.minor.micro) should match with runtime version. To correct, install the correct version of IDE or Runtime. |
| E30 | EtherCAT communication failure during operational mode. | EtherCAT network operation failed due to a network communication error. | Check the EtherCAT network wiring and devices state. Re-start the application. |
| E31 | EtherCAT communication failure during preop mode. | EtherCAT network operation failed due to a network communication error. | Check the EtherCAT network wiring and devices state. Re-start the application. |
| E32 | EtherCAT communication failure during bootstrap mode. | EtherCAT network operation failed due to a network communication error. | Check the EtherCAT network wiring and devices state. Re-start the application. |
| E33 | EtherCAT failed to initialize into operational mode. | EtherCAT network initialization failed due to a network initialization or communication error. | Check the EtherCAT network wiring and devices state. Re-start the application. |
| E34 | EtherCAT failed to initialize into preop mode. | EtherCAT network initialization failed due to a network communication error. | Check the EtherCAT network wiring and devices state. Re-start the application. |
| E35 | EtherCAT failed to initialize into bootstrap mode. | EtherCAT network initialization failed due to a network communication error. | Check the EtherCAT network wiring and devices state. Re-start the application. |

| Error | Description | Cause | Remedy |
|-------|--|--|--|
| E36 | EtherCAT network discovery failed. | EtherCAT network discovery failed to detect EtherCAT devices or EtherCAT network discovery detected changing network topology (due to intermittent connections) during the scanning process. | Check the power state of EtherCAT devices and EtherCAT connection LEDs. Verify the condition of the power and EtherCAT cables and their connections. If all cabling and connections are good, isolate which EtherCAT device is causing the problem and check its operating state according to manufacturer instructions. |
| E37 | EtherCAT failed to return to init state. | EtherCAT network initialization failed due to a network communication error. | Check the EtherCAT network wiring and devices state. Re-start the application. |
| E50 | Backup to SD card failed. | An unrecoverable error occurred during the backup operation. | Repeat the backup to SD card operation. If it fails again, replace the SD card. |
| E51 | Restore from SD card failed.. | An unrecoverable error occurred during the restore operation. | Do NOT reboot the PDMM! Repeat the restore operation. If it fails again, reset the PDMM to factory defaults. If problem persists, return hardware for repair. |
| E52 | SD Backup files are missing or corrupt. | The restore operation failed due to missing, incomplete, or corrupt files on the SD card. | Perform a backup operation before the restore or use an SD card with valid backup files. |
| E53 | SD Backup files are not compatible. | The restore operation failed. The backup files are not compatible with the PDMM 800 MHz model. | Use an SD card with a backup from a PDMM 800 MHz model. |

15.3.3 Alarms

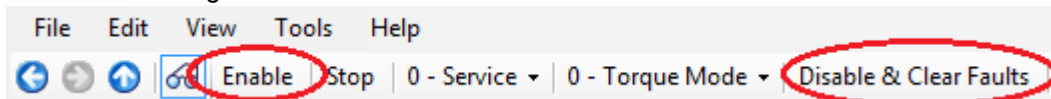
| Alarm | Description | Cause | Remedy |
|-------|---|---|---|
| A01 | High temperature exceeded | CPU temperature near the safe operating temperature limit. | Check airflow and operating environment are within hardware specifications. |
| A02 | Low on memory. | Memory leak or corruption. | Power-off/on. If problem is recurrent, check release notes for firmware updates or return hardware for repair. |
| A04 | Low input voltage | +24 volt input power is +19 volts or less. | Check power supply voltage and connection to the PDMM. |
| A12 | Flash memory is low on free space. | Flash memory is almost full. | Clean-up the flash memory by removing log files, application programs, recipes, or other data files. Reset to factory defaults. |
| A21 | Recoverable process or thread failed to respond during operation. | KAS non-runtime code failed during normal operation and was automatically restarted. | If problem is recurrent, power-off/on. Check release notes for firmware updates. |
| A23 | CPU is overloaded | | Reduce the sample rate, simplify the application, or reduce the application cycles. |
| A30 | EtherCAT missed a send frame during operation mode. | EtherCAT master was unable to send a frame for one or more cycles. | Reduce the controller CPU load. |
| A38 | EtherCAT missed a receive frame during operation mode. | EtherCAT master did not receive a frame for one or more cycles. | Check the EtherCAT network wiring and devices. |
| A40 | Local digital IO missed a cyclic update | Local digital IO was not updated during a cycle or the updates are no longer synchronous. | Reduce the sample rate, simplify the application, or reduce the application cycles. |
| A53 | AKD PDMM-M-MC was replaced with the higher performance M1 model. | AKD PDMM-M-M1 model was restored using backup files from a AKD PDMM-M-MC model. | Replace the AKD PDMM with an 800MHz model or use backup files from higher performance AKD PDMM-M-M1 model.. |

15.4 Clearing Faults

You can clear drive faults in three ways: 1. using the EWV¹ toolbar; 2. using the EWV Faults screen, 3. using an external digital signal. Clearing the fault automatically disables the drive (by issuing the DRV.DIS command). After the fault is cleared, you must then enable the drive again (by issuing the DRV.EN command). Using external digital signal, clearing the fault does not automatically disable the drive. Once the fault is cleared, the drive will immediately become enabled again when no fault conditions exist.

The external digital signal method is useful when the application does not require a field bus, therefore the controller cannot issue fieldbus commands (for example, DRV.EN). The fault clearing function is simpler in this case. The controller can clear a fault condition with a single output and no fieldbus command is required. The hardware enable input is still utilized for safety. Many machines are set to disable when a fault occurs, therefore, the enable circuit is connected only after the fault clear circuit is toggled.

1. Using the EWV toolbar. Click the **Clear Fault** button to clear the fault, then click the **Enable** button to enable the drive again.



2. Using an ASCII command. In EWV terminal screen, enter DRV.CLRFAULTS, then enter DRV.EN to enable the drive.
3. Using an external digital signal. First, provide a disable command to the drive, since this command is not automatically issued when the fault is cleared using an external digital signal. You must provide this disable command after the drive has disabled itself and before the clear fault command is issued in order for the clear fault command to be accepted.

Digital input mode 1 clears a fault condition. In the EWV terminal screen, enter DINx.MODE 1, with x set to the corresponding digital input number. See the AKD PDMM Installation Manual (Digital Inputs (X7/X8) (→ p. 1)) for digital input numbers and corresponding pin connections.

Example:

DIN2.MODE 1 sets digital input 2 (pin 9 on connector X7) to perform the clear fault function. Any free digital input shown in Table x can be set to mode 1 (Reset faults).

15.5 Parameter and Command Error Messages

| Error Message | Meaning | Remedy | Occurrence |
|--|--|---|----------------------|
| Error: Parameter or command was not found. | Parameter or command string was not recognized as a known command. | Check the entered string for accuracy. Consult the reference guide or enter DRV.LIST in the EWV ² terminal screen to find valid parameters and commands. | General. |
| Error: Not a trigger parameter. | Parameter cannot be used for triggering the scope. | Use a different parameter as a trigger value. | Recorder parameters. |
| Error: Parameter not recordable. | Parameter cannot be scoped. | Do not scope this parameter. | Recorder parameters. |

¹Embedded Workbench Views

²Embedded Workbench Views

| Error Message | Meaning | Remedy | Occurrence |
|---|--|--|---|
| Error: Wrong argument for parameter or command. | Argument was not accepted. | Check the entered argument for accuracy. Consult the reference guide or enter DRV.LIST in the EWW terminal screen to find valid parameter and command arguments. | General. |
| Error: No arguments allowed for parameter or command. | Parameter or command supports no arguments. | Enter parameter or command again with no arguments. | Commands and read-only parameters |
| Error: Too many arguments for parameter or command. | Too many arguments provided for this parameter or command. | Enter again with fewer arguments. | All parameters and commands with no or one argument. |
| Error: No float allowed. | Float number was entered, but only integer numbers are allowed. | Enter integer value instead. | All parameters and commands with integer arguments. |
| Error: Parameter or command is read-only. | Argument was not accepted because the parameter or command is read-only. | Enter again with no arguments. | Commands and all read-only parameters. |
| Error: Parameter is temporarily write protected. | Argument was not accepted because the parameter is currently read-only. | Check reference guide or KAS IDE help to determine if the parameter is read-only, read-write, or nonvolatile. | Some parameters that come from the Smart Feedback Device (SFD). |
| Error: Argument greater than maximum value. | Entered argument was beyond maximum value. | Enter a value that is within the allowable range. See reference guide or KAS IDE help for ranges. | Commands and read-write parameters. |
| Error: Too many characters in parameter or command. | Parameter or command string was too long. | Shorten string and enter again. | General. |
| Error: Argument less than minimum value. | Entered argument was too small. | Enter value that is within the limits. | Commands and read-write parameters. |
| Error: Argument is out of data range. | Entered argument was not within the data range limits. | Enter value that is within the limits. | Commands and read-write parameters. |
| Error: Parameter or command is password protected. | Use of this parameter or command requires a password. | Contact technical support to obtain password. Enter valid password. | Password protected parameters and commands. |
| Error: Wrong operation mode for parameter or command. | Requested parameter or command cannot be performed in this operation mode. | Change to valid operation mode. | |

| Error Message | Meaning | Remedy | Occurrence |
|---|--|--|---|
| Error: Drive enabled. Disable drive and continue. | Action can only be performed if drive is disabled. | Disable drive and repeat action. | |
| Error: Drive disabled. Enable drive and continue. | Action can only be performed if drive is enabled. | Enable drive and repeat action. | |
| Error: Data value is invalid. | Entered data was not valid. | Provide valid data. | |
| Error: Argument must be an even number. | Argument is not an even number. | Enter a value divisible by two without remainder. | |
| Error: Argument must be an odd number. | Argument must be an odd number. | Enter a value not divisible by two with remainder. | |
| Error: Axis is not homed. | Axis is not homed. | Home axis and repeat action. | |
| Error: All recorder channels empty. | No data is specified to record. | Specify the data to be recorded and start again. | Recorder commands. |
| Error: Process already active. | Required action is already running. | Wait until action is finished and start again if needed. | |
| Error: Motor is currently in motion. | Action can only be performed if motor is not moving. | Stop motion and repeat action. | Motion tasks, service motion, and others. |
| Error: EEPROM is bad or does not exist. | EEPROM is damaged. | Restart drive. If fault persists, replace the drive and consult technical support. | |
| Error: Unknown board. | | | |
| Error: Fault exists. Correct fault condition and then clear faults. | A fault is present in the system. | Clear the fault, disable the drive, and then enable the drive again. | |
| Error: Cannot clear faults. Issue software or hardware disable first. | | Clear hardware or software enable and repeat action. | |
| Error: EEPROM is busy. | | | |
| Error: Invalid motor/feedback poles ratio. | | | |
| Error: Unknown Fault. | Situation should not happen. | Clear fault. If fault recurs, consult technical support. | Unknown. |
| Error: Invalid motor/feedback poles ratio. | | | |

| Error Message | Meaning | Remedy | Occurrence |
|--|---------|--------|------------|
| Error: Invalid motion task parameter. | | | |
| Error: Invalid motion task number. | | | |
| Error: Invalid motion task velocity. | | | |
| Error: Invalid motion task acceleration or deceleration. | | | |
| Error: Invalid motion task customer profile table number. | | | |
| Error: Invalid motion task following number. | | | |
| Error: Function not available for the active command source. | | | |
| Error: Invalid Bode plot mode for this function. | | | |
| Error: Invalid sine-sweep Bode plot mode. | | | |
| Error: Bode plot start frequency greater than or equal to end frequency. | | | |
| Error: Function not available while a controlled stop is in process. | | | |
| Error: Not enough memory available. | | | |
| Error: Function not available while a Bode-plot measurement is active. | | | |
| Error: Iu/Iv offset calculation not finished. | | | |
| Error: Buffer overflow. | | | |
| Error: Cannot save to EEPROM while inrush relay is closed. | | | |

| Error Message | Meaning | Remedy | Occurrence |
|--|---------|--------|------------|
| Error: Test mode is off. | | | |
| Error: Cannot change digital input mode. Issue software or hardware disable first. | | | |
| Error: Internal drive procedure active: controlled stop, burn-in, phase find, or zero. | | | |
| Error: General motion fault. | | | |

15.6 Unknown Fault

This fault message occurs when an undefined fault condition is encountered.

15.6.1 Remedies

1. Click **Clear Fault**.

16 Troubleshooting the AKD PDMM

Drive problems occur for a variety of reasons, depending on the conditions in your installation. The causes of faults in multi-axis systems can be especially complex. If you cannot resolve a fault or other issue using the troubleshooting guidance presented below, customer support can give you further assistance.

Eliminate errors and faults in compliance with work safety rules. Troubleshooting only by qualified and trained staff.

More details on the removal of faults can be found in the [EWV¹](#) online help and in [KDN](#).

| Problem | Possible Causes | Remedy |
|-------------------------------------|---|---|
| HMI message: Communication fault | <ul style="list-style-type: none"> wrong cable used, cable plugged into wrong position on drive or PC wrong PC interface selected | <ul style="list-style-type: none"> plug cable into the correct sockets on the drive and PC select correct interface |
| Drive does not enable | <ul style="list-style-type: none"> HW Enable not wired HW or SW Enable not set | <ul style="list-style-type: none"> connect HW Enable (X8 pin 4) Apply 24V to HW Enable and select SW Enable in EWV / Fieldbus |
| Motor does not rotate | <ul style="list-style-type: none"> drive not enabled software enable not set break in setpoint cable motor phases swapped brake not released drive is mechanically blocked motor pole no. set incorrectly feedback set up incorrectly | <ul style="list-style-type: none"> apply ENABLE signal set software enable check setpoint cable correct motor phase sequence check brake control check mechanics set motor pole no. set up feedback correctly |
| Motor oscillates | <ul style="list-style-type: none"> gain is too high (speed controller) feedback cable shielding broken AGND not wired up | <ul style="list-style-type: none"> reduce VL.KP (speed controller) replace feedback cable join AGND to CNC-GND |
| Drive reports following error | <ul style="list-style-type: none"> I_{rms} or I_{peak} set too low current or velocity limits apply accel/decel ramp is too long | <ul style="list-style-type: none"> verify motor/drive sizing verify that IL.LIMITN/P, VL.LIMITN/P are not limiting the drive reduce DRV.ACC/DRV.DEC |
| Motor overheating | <ul style="list-style-type: none"> motor operating above its rating motor current settings incorrect | <ul style="list-style-type: none"> verify motor/drive sizing verify motor continuous and peak current values are set correctly |
| Drive too soft | <ul style="list-style-type: none"> K_p (speed controller) too low K_i (speed controller) too low filters set too high | <ul style="list-style-type: none"> increase VL.KP (speed controller) increase VL.KI (speed controller) refer to documentation regarding reducing filtering (VL.AR*) |
| Drive runs roughly | <ul style="list-style-type: none"> K_p (speed controller) too high K_i (speed controller) too high filters set too low | <ul style="list-style-type: none"> reduce VL.KP (speed controller) reduce VL.KI (speed controller) refer to documentation regarding increasing filtering (VL.AR*) |

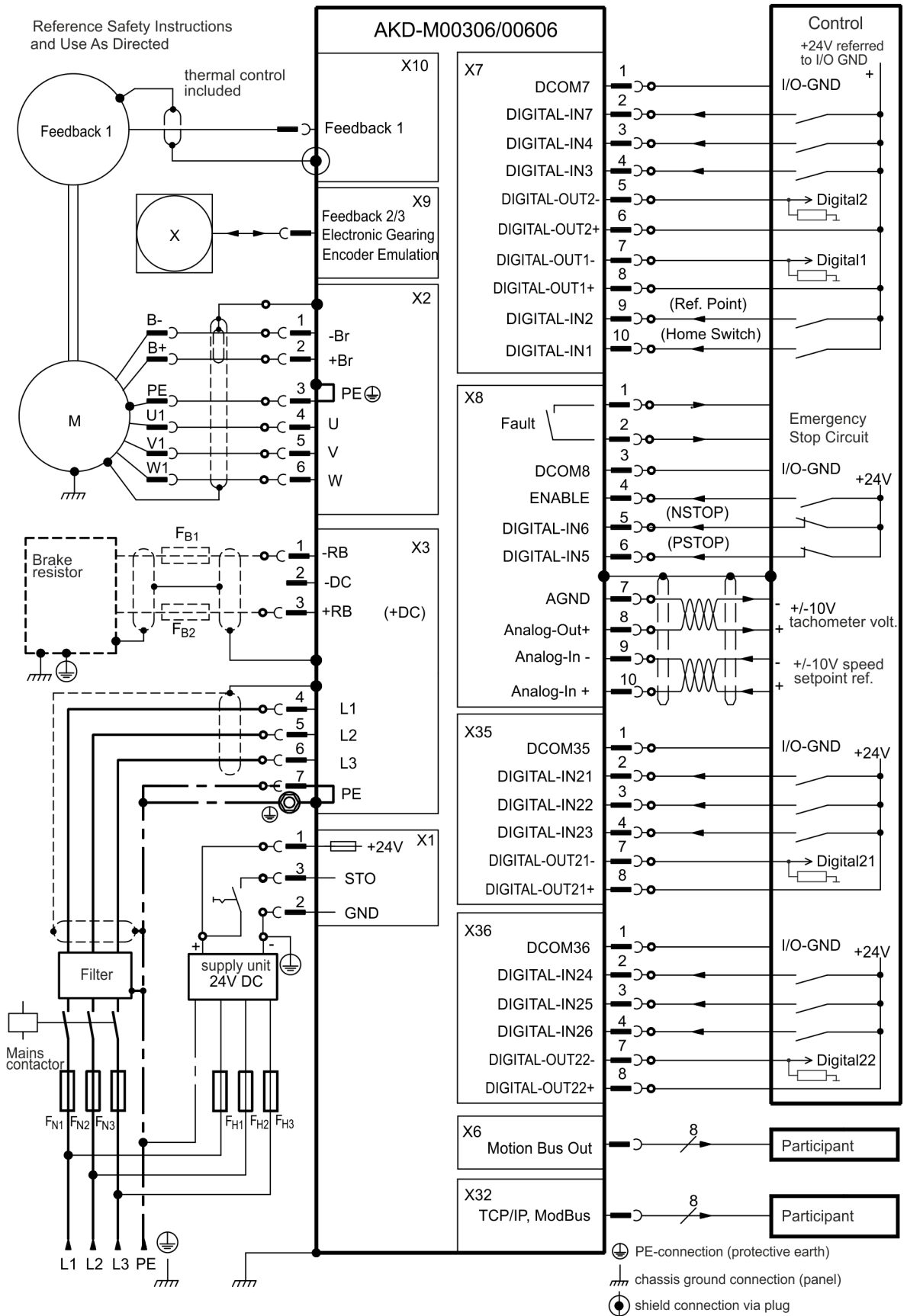
¹Embedded Workbench Views

| Problem | Possible Causes | Remedy |
|--|--|--|
| During installation, the message "Please wait while the installer finishes determining your disk space requirements" appears and never disappears. | <ul style="list-style-type: none"><li data-bbox="533 226 783 255">• MSI installer issue.<li data-bbox="533 320 900 349">• Harddisk space not sufficient | <ul style="list-style-type: none"><li data-bbox="1005 226 1449 349">• Cancel the installation and relaunch the installer (you may need to try several times, the problem is random).<li data-bbox="1005 353 1410 450">• Make sure that you have enough disk space on your hard disk (~500MB). |

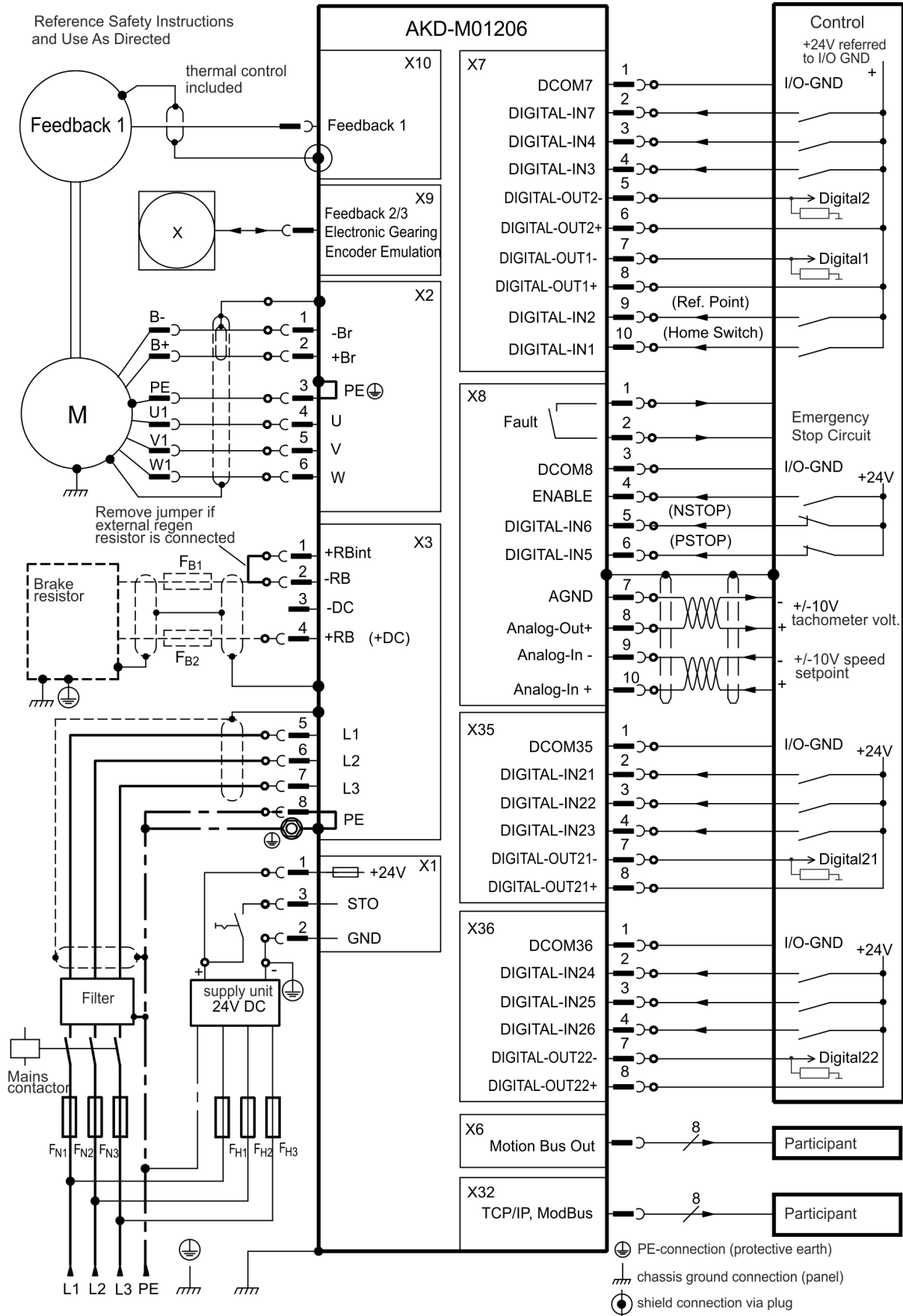
17 Connection Diagrams

| | | |
|-------|---|-----|
| 17.1 | Connection diagram AKD-M00306, AKD-M00606 | 207 |
| 17.2 | Connection diagram AKD-M01206 | 208 |
| 17.3 | Connection diagram AKD-M02406, AKD PDMM-M00307 to AKD PDMM-M02407 | 209 |
| 17.4 | 24 V auxiliary supply (X1) | 210 |
| 17.5 | Motor Power Connection (X2) | 211 |
| 17.6 | External regen resistor (X3) | 214 |
| 17.7 | DC Bus link (X3, X14) | 216 |
| 17.8 | Mains supply connection (X3, X4) | 219 |
| 17.9 | I/O Connection | 222 |
| 17.10 | Analog Output (X8) | 225 |
| 17.11 | Analog Input (X8) | 226 |
| 17.12 | Command encoder signal connection | 227 |
| 17.13 | Pulse / Direction signal connection | 229 |
| 17.14 | CW / CCW signal connection | 230 |
| 17.15 | Feedback connector (X10) | 231 |
| 17.16 | Feedback connector (X9) | 232 |
| 17.17 | Feedback connector (X7) | 233 |

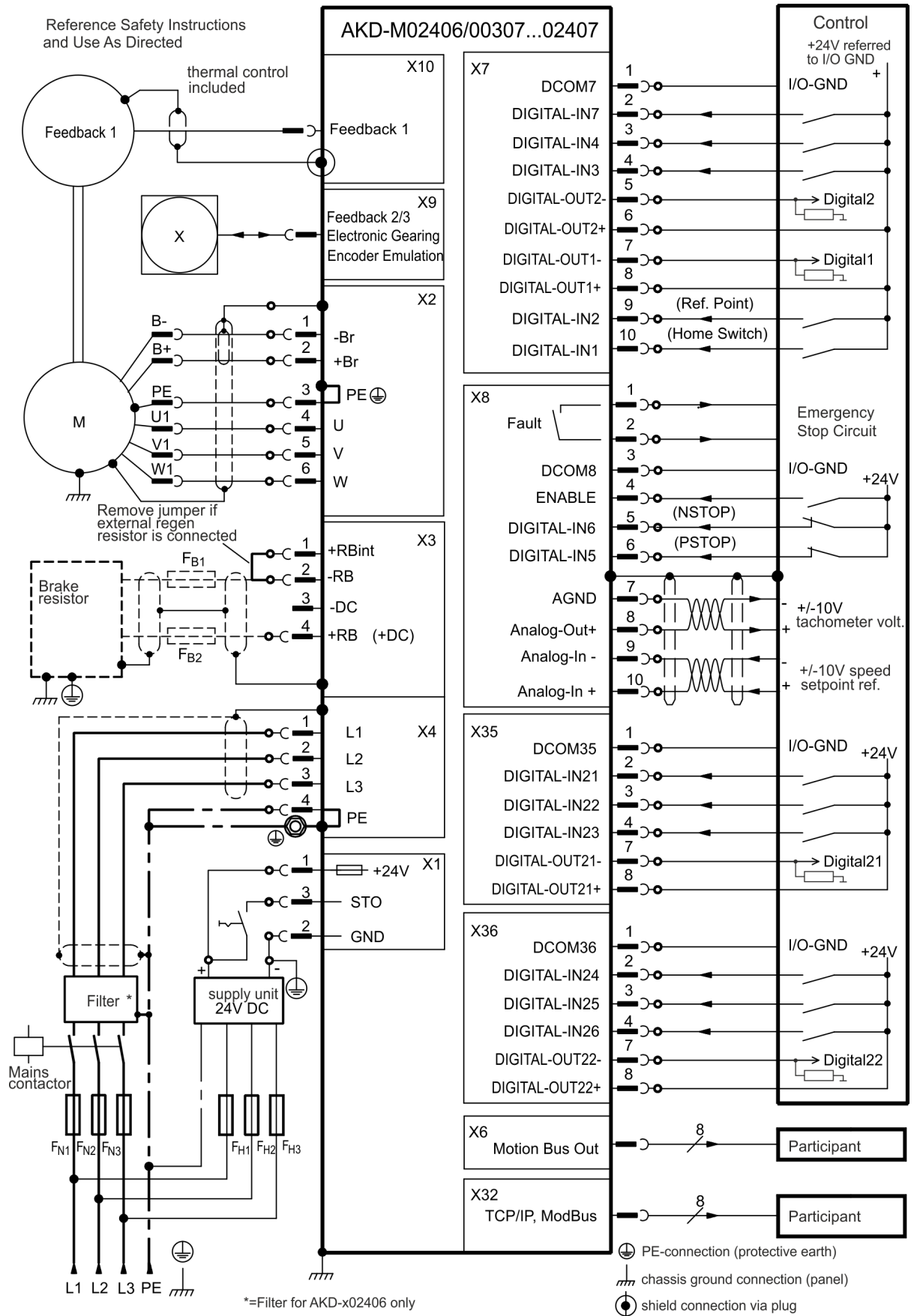
17.1 Connection diagram AKD-M00306, AKD-M00606



17.2 Connection diagram AKD-M01206

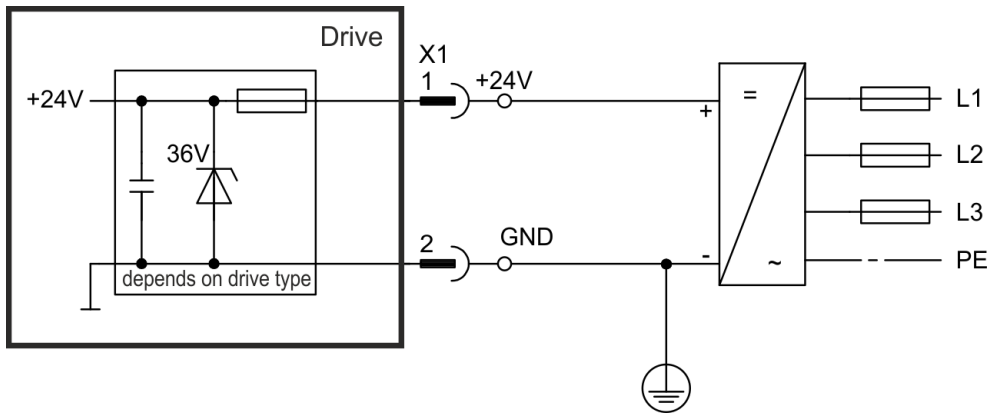


17.3 Connection diagram AKD-M02406, AKD PDMM-M00307 to AKD PDMM-M02407



17.4 24 V auxiliary supply (X1)

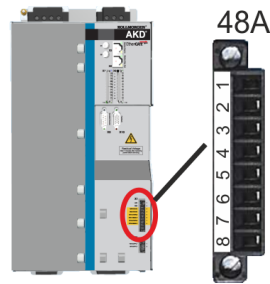
The following diagram describes external 24 VDC power supply, electrically isolated, for example, via an isolating transformer. The required current rating depends on the use of motor brake and option card .



17.4.1 AKD-M003 to 024, connector X1

| Pin | Signal | Description |
|-----|--------|------------------------------|
| 1 | +24 | +24 VDC Auxiliary voltage |
| 2 | GND | 24V Supply GND |
| 3 | STO | STO enable (Safe Torque Off) |

17.4.2 AKD-M048, connector X1



| Pin | Signal | Description |
|-----|--------------|----------------------------------|
| 1 | +24 V | +24 VDC Auxiliary voltage |
| 2 | GND | 24V Supply GND |
| 3 | STO +24V | STO+24 VDC Auxiliary voltage |
| 4 | STO GND | STO Supply GND |
| 5 | STO-Status 1 | Safe Torque Off status channel 1 |
| 6 | STO-Enable 1 | Safe Torque Off enable channel 1 |
| 7 | STO-Status 2 | Safe Torque Off status channel 2 |
| 8 | STO-Enable 2 | Safe Torque Off enable channel 2 |

17.5 Motor Power Connection (X2)

Together with the motor supply cable and motor winding, the power output of the drive forms an oscillating circuit. Characteristics such as cable capacity, cable length, motor inductance, and frequency determine the maximum voltage in the system.

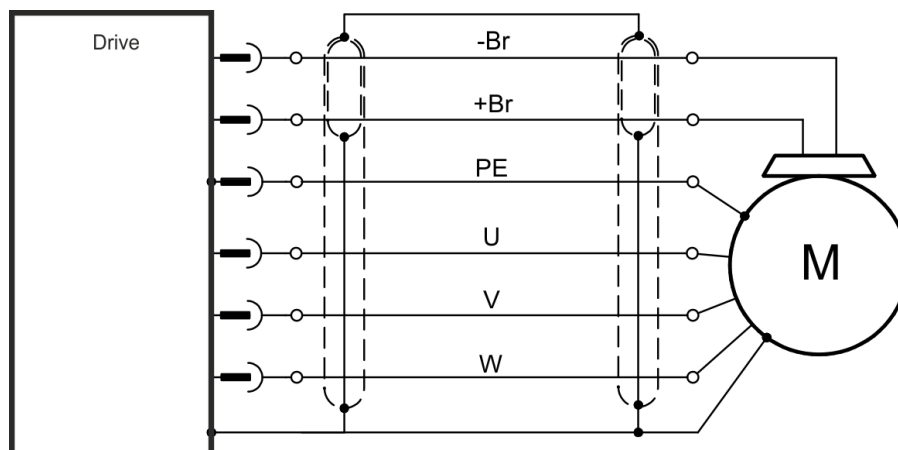
The AKD PDMM drive is able to protect the connected motor from overloading, if the parameters are set correctly and the thermal protection sensor is connected and supervised. With Kollmorgen motors the valid data are automatically set by the internal motor database. With motors from other manufacturers the data from the nameplate must be entered to the referring fields in the motor view of the Kollmorgen setup software [EWV](#)¹.

The dynamic voltage rise can lead to a reduction in the motor operating life and, on unsuitable motors, to flashovers in the motor winding.

- Only install motors with insulation class F (acc. to IEC60085) or above.
- Only install cables that meet the requirements .

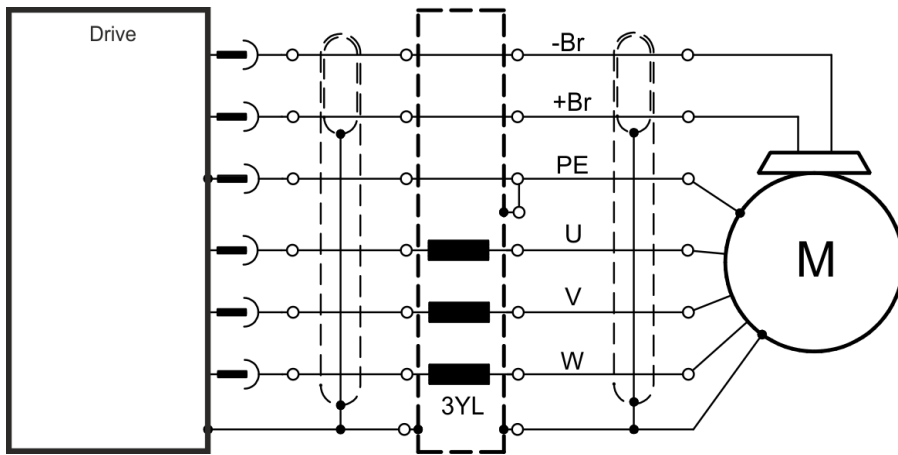
With long motor cables leakage currents endanger the output stage of the drive. For cable lengths from 25 m to 50 m, a motor choke (3YL) must be wired into the motor cable (near the drive). Refer to your regional Accessories Manual for Kollmorgen chokes.

Cable length \leq 25 m



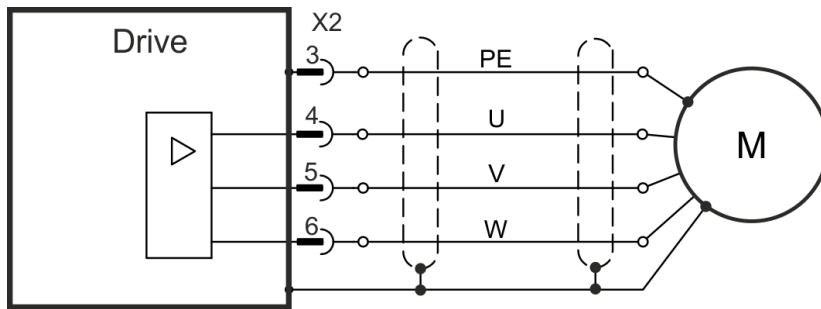
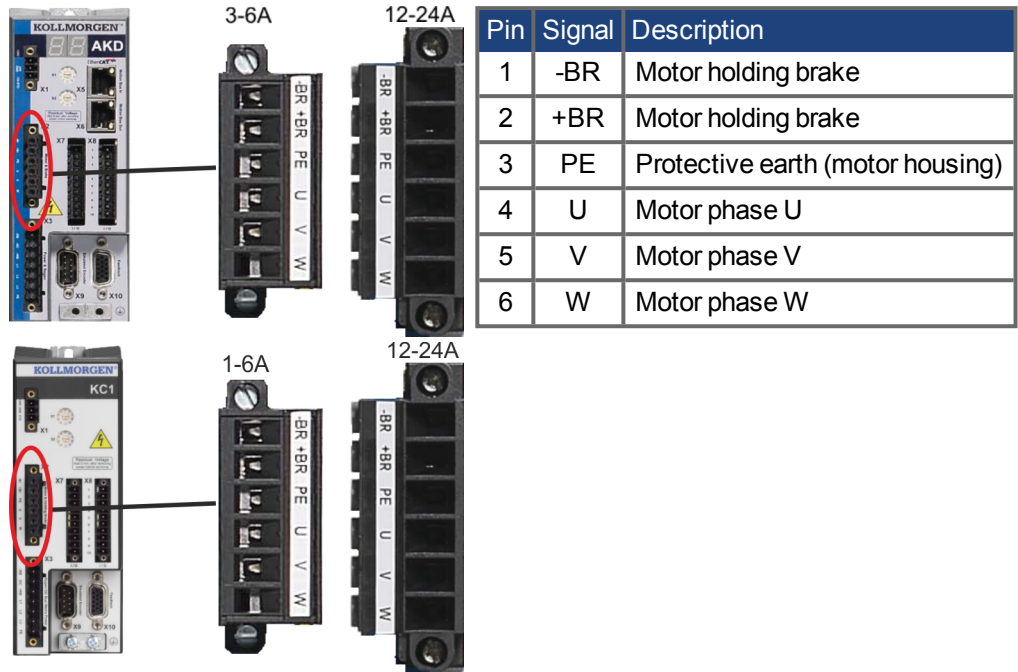
Cable length $>$ 25 m

¹Embedded Workbench Views



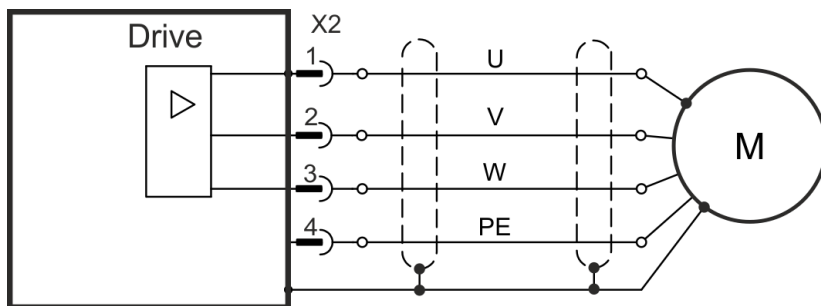
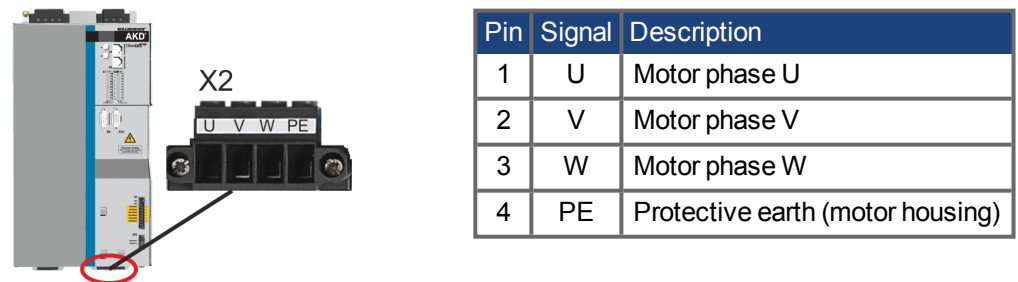
17.5.1 AKD-M003 to 024, power connector X2

Power connector X2



17.5.2 AKD-M048, power connector X2

Cable length maximum 25 m.

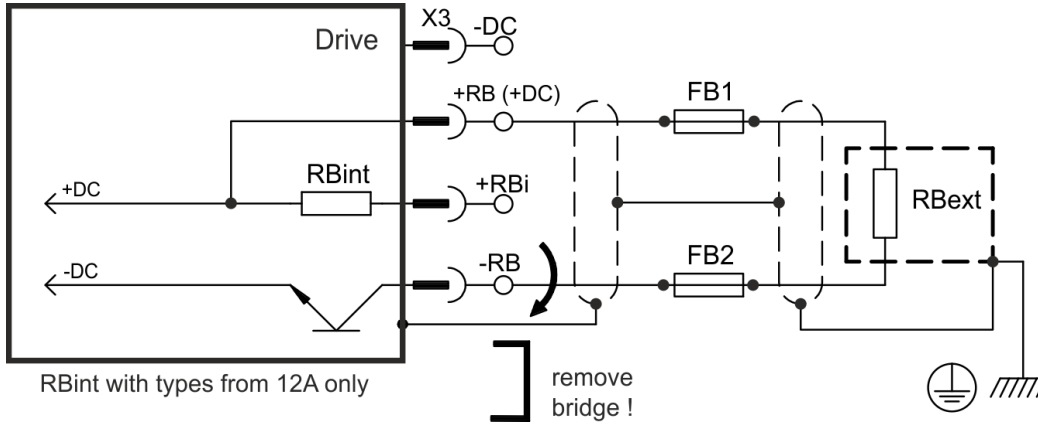


17.6 External regen resistor (X3)

For technical data on the brake circuit .

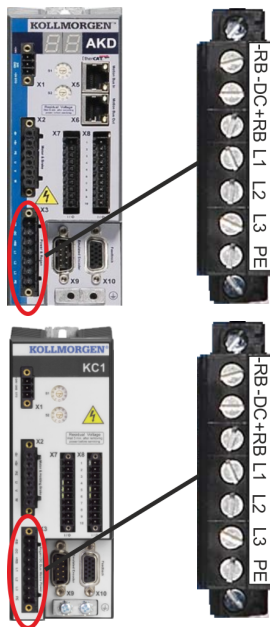
17.6.1 AKD-M003 to 024, regen connector X3

Fusing (such as fusible cut-outs) to be provided by the user .

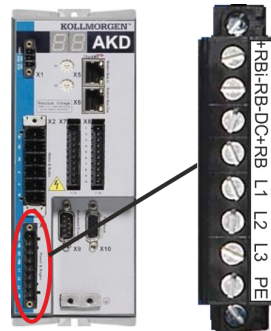


FB1 / FB2 fusing

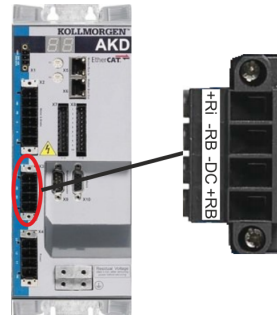
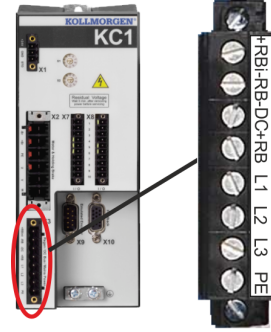
| Drive Model | Ampere rating@230V | Ampere rating@480V | UL region example: | CE Region example (Siba): |
|-----------------|--------------------|--------------------|---------------------|---------------------------|
| AKD-M003 to 012 | 10A | 40A | Bussmann FWP-xxA14F | 110V to 400V: gRL |
| AKD-M024 | 15A | 50A | | 400V to 480V: aR |



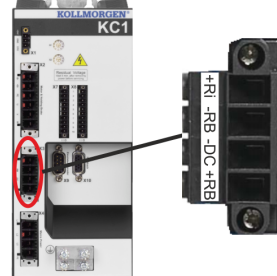
| AKD-M0010600306 to AKD-M00606 (X3) | | |
|------------------------------------|--------|----------------------------------|
| Pin | Signal | Description |
| 1 | -RB | External Regen Resistor negative |
| 3 | +RB | External Regen Resistor positive |



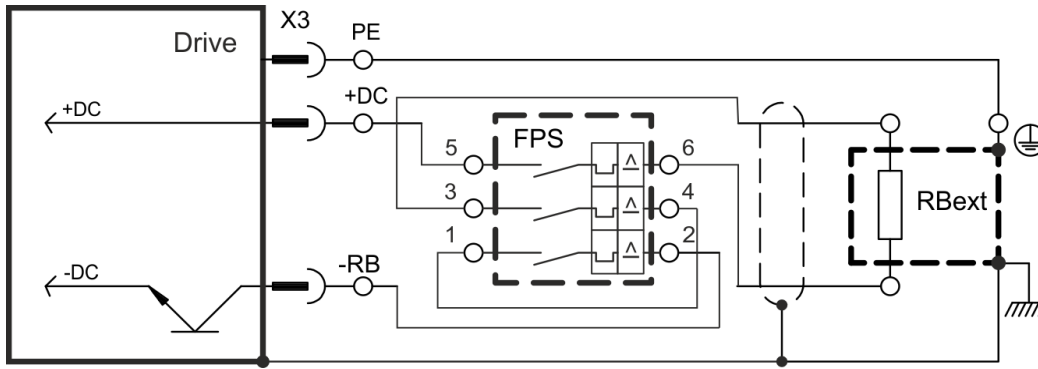
| AKD-M1206 (X3) | | |
|----------------|--------|----------------------------------|
| Pin | Signal | Description |
| 1 | +Rbint | Internal Regen Resistor positive |
| 2 | -RB | External Regen Resistor negative |
| 4 | +RB | External Regen Resistor positive |



| AKD-M02406 & AKD-M-00307 to 02407 (X3) | | |
|--|--------|----------------------------------|
| Pin | Signal | Description |
| 2 | -RB | External Regen Resistor negative |
| 4 | +RB | External Regen Resistor positive |

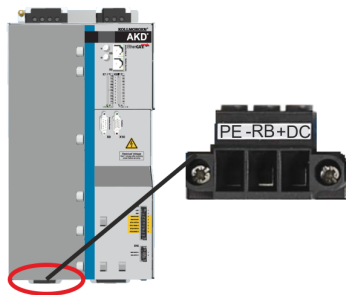


17.6.2 AKD-M048, regen connector X3



FPS: Fritzen DC Powerswitch

| RBext | FPS | Range [In] | cable cross section |
|---------------|--------|------------|--------------------------|
| BAS(U)2000-10 | FPS-16 | 10 to 16 A | min. 2.5 mm ² |
| BAS(U)3000-10 | FPS-20 | 16 to 20 A | min. 4 mm ² |
| BAS(U)6000-10 | FPS-25 | 20 to 25 A | min. 4 mm ² |



| AKD-M04807 | | |
|------------|--------|----------------------------------|
| Pin | Signal | Description |
| 1 | PE | Protective Earth |
| 2 | -RB | External Regen Resistor negative |
| 3 | +DC | External Regen Resistor positive |

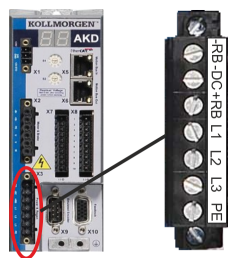
17.7 DC Bus link (X3, X14)

The DC bus link can be connected in parallel so that the regen power is divided between all the drives that are connected to the same DC bus link circuit. Every drive must have its own power connection to mains voltage, even if the DC bus link is used. Drives working regeneratively very often should be placed beside drives, which need energy. That reduces current flow on longer distances.

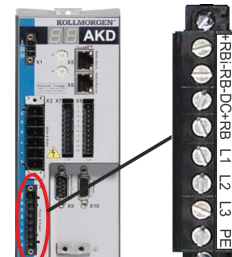
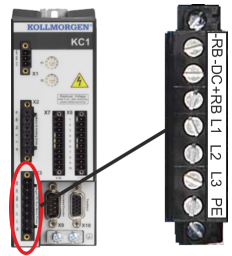
- The sum of the rated currents for all of the drives connected in parallel to an **AKD-M001003 to 024** must not exceed 48 A. Use 6 mm², unshielded single cores with a max. length of 200 mm; use 6 mm² shielded cables for longer lengths. In this case no fuse for line protection is required.
- The sum of the rated currents for all of the **AKD-M048** connected in parallel to an AKD-M048 must not exceed 96 A. AKD-M048 should be connected in parallel only with AKD-M048 drives. Use 16 mm², unshielded single cores with a max. length of 30 mm; use 16 mm² shielded cables for longer lengths.

The drives can be destroyed, if DC bus link voltages are different. Only drives with mains supply from the same mains (identical mains supply voltage) may be connected by the DC bus link. DC bus link connection of AKD-M048 to other drives than AKD-M048 is not allowed.

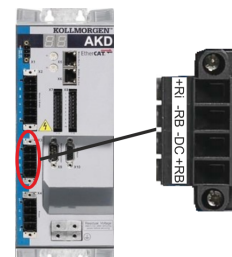
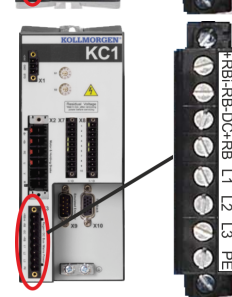
The phase lost/ main control is not working on DC-Bus connected drives. A phase lost on a drive inside a DC Bus connection will be not detected. External phase monitoring to prevent overload of the rectifier is recommended.



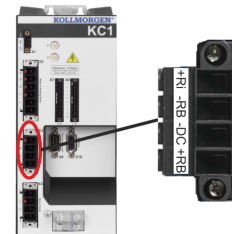
| AKD-M0010600306 to AKD-M00606 (X3) | | |
|------------------------------------|-----------|----------------------|
| Pin | Signal | Description |
| 2 | -DC | DC-Link Bus negative |
| 3 | +DC (+RB) | DC-Link Bus positive |

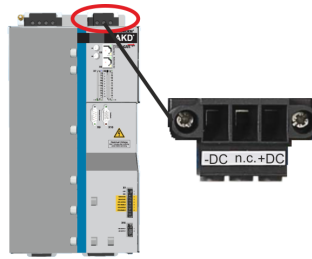


| AKD-M01206 (X3) | | |
|-----------------|-----------|----------------------|
| Pin | Signal | Description |
| 3 | -DC | DC-Link Bus negative |
| 4 | +DC (+RB) | DC-Link Bus positive |



| AKD-M02406 & AKD PDMM00307 to AKD PDMM02407 (X3) | | |
|--|-----------|----------------------|
| Pin | Signal | Description |
| 3 | -DC | DC-Link Bus negative |
| 4 | +DC (+RB) | DC-Link Bus positive |

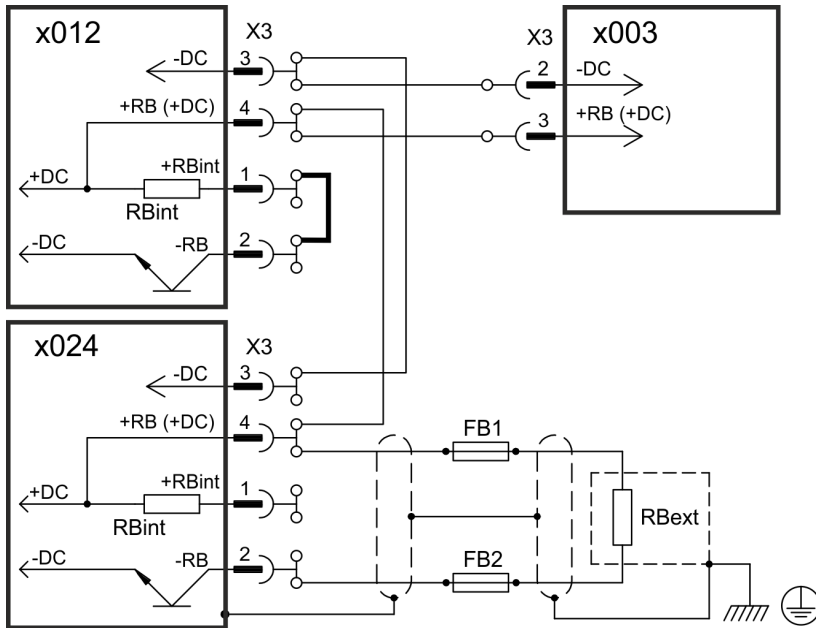




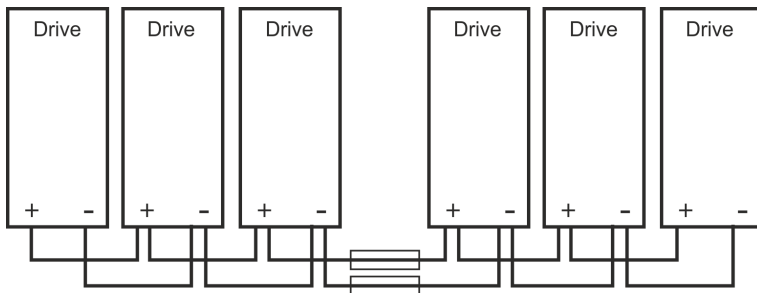
| AKD-M04807 (X14) | | |
|------------------|--------|----------------------|
| Pin | Signal | Description |
| 1 | -DC | DC-Link Bus negative |
| 2 | n.c. | not connected |
| 3 | +DC | DC-Link Bus positive |

NOTE: Should be connected in parallel only with AKD-M048 drives.

17.7.1 DC Bus topology with Y connectors (24 A max.)

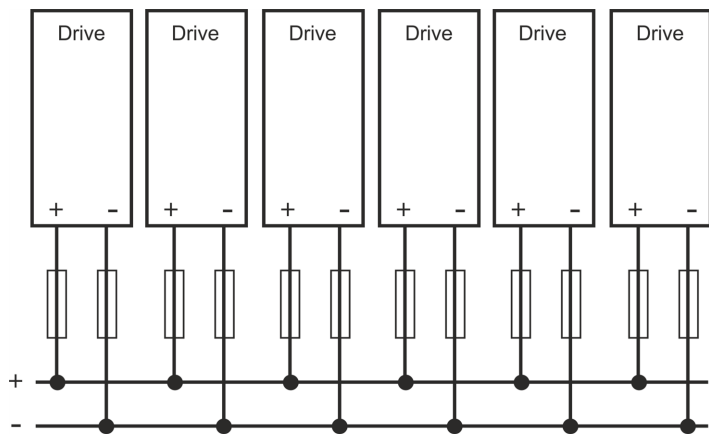


Without DC Bus fuses, other devices can become damaged or destroyed if, for example, a device fails due to an internal short circuit. If multiple drives are connected in parallel, then it is usual to insert DC Bus fuses between groups of drives (with a group consisting of two or three devices, depending on the current) in order to limit any possible resulting damage. Fuses cannot avoid damage by current peaks completely.



17.7.2 DC Bus topology with busbar

This wiring does not require Y connectors. If a device fails due to a short-circuit, only its DC Bus fuses are tripped and the rest of the network continues uninterrupted. The solid busbars can conduct significantly larger currents, because the compensating current does not flow through the connector as above. For this reason, almost as many drives as desired can be connected in parallel in this form. This arrangement is frequently also useful for connecting KCM capacitor modules.

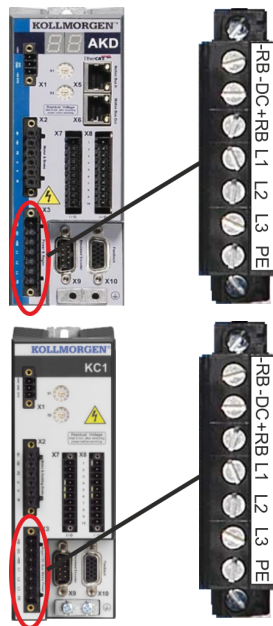


17.8 Mains supply connection (X3, X4)

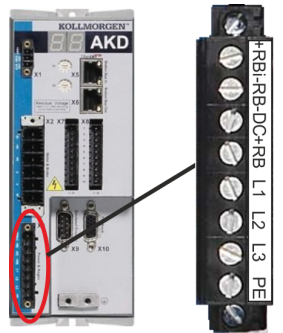
Drives in the AKD PDMM series can be supplied as follows:

- AKD-Mzzz06: 1 or 3 phase industrial supply networks (not more than 200 kA symmetrical rated current at 120 V and 240 V).
- AKD-Mzzz07: 3 phase industrial supply networks (not more than 200 kA symmetrical rated current at 240 V, 400 V and 480 V).
- AKD-M04807: In case of mains voltage asymmetry >3% a mains choke 3L0,24-50-2 (see regional Accessories Manual) must be used.

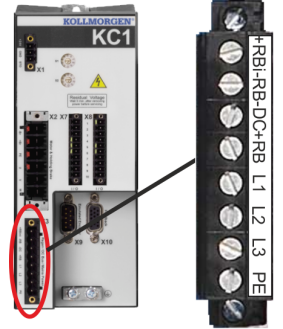
Connection to other voltage types of supply networks is possible with an additional isolating transformer. Periodic overvoltages between phases (L1, L2, L3) and the housing of the drive must not exceed 1000 V peak. In accordance with IEC 61800, voltage spikes (< 50 μs) between phases must not exceed 1000 V. Voltage spikes (< 50 μs) between a phase and the housing must not exceed 2000 V.



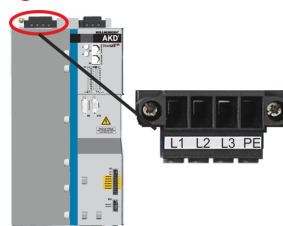
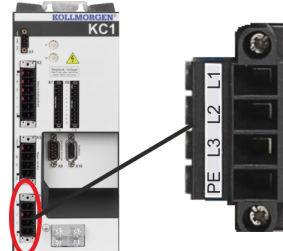
| AKD-M0010600306 to AKD-M00606 (X3) | | |
|------------------------------------|--------|------------------|
| Pin | Signal | Description |
| 4 | L1 | Line 1 |
| 5 | L2 | Line 2 |
| 6 | L3 | Line 3 |
| 7 | PE | Protective Earth |



| AKD-M01206 (X3) | | |
|-----------------|--------|------------------|
| Pin | Signal | Description |
| 5 | L1 | Line 1 |
| 6 | L2 | Line 2 |
| 7 | L3 | Line 3 |
| 8 | PE | Protective Earth |



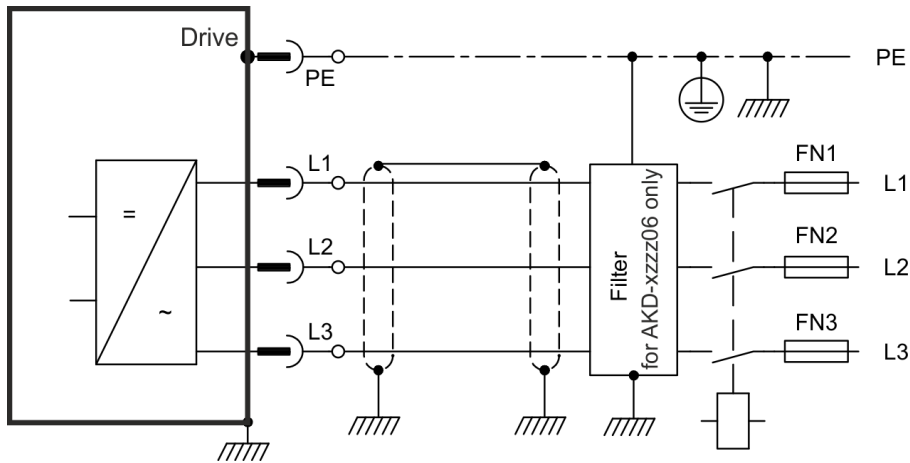
| AKD-M02406 & AKD-M00307 to AKD-M02407 (X4) | | |
|--|--------|------------------|
| Pin | Signal | Description |
| 1 | L1 | Line 1 |
| 2 | L2 | Line 2 |
| 3 | L3 | Line 3 |
| 4 | PE | Protective Earth |



| AKD-M04807 (X4) | | |
|-----------------|--------|------------------|
| Pin | Signal | Description |
| 1 | L1 | Line 1 |
| 2 | L2 | Line 2 |
| 3 | L3 | Line 3 |
| 4 | PE | Protective Earth |

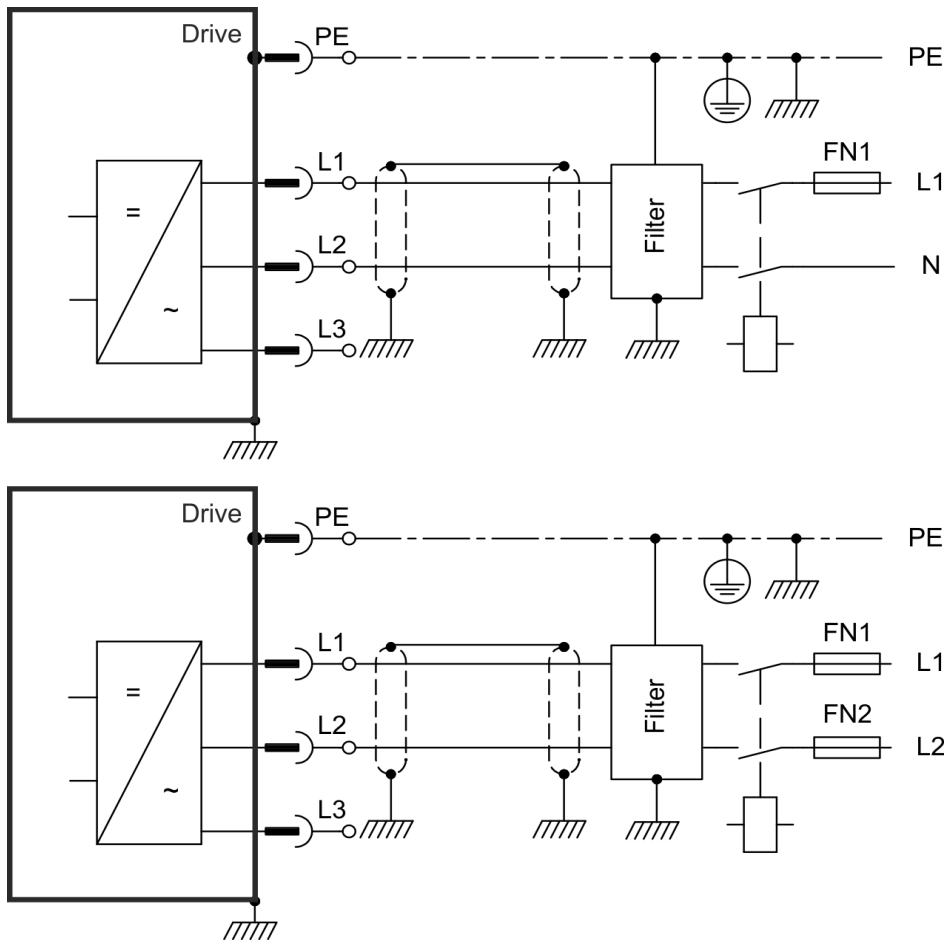
17.8.1 Three phase connection (all AKD PDMM types)

- Directly to 3-phase supply network, supply networks
- Filtering for AKD-Mzzz06 to be provided by the user.
- Fusing (such as fusible cut-outs) to be provided by the user .



17.8.2 Single/Dual phase connection (AKD-M0010600306 to AKD-M01206 only)

- Directly to single-phase supply network (120 V_{-10%} to 240 V_{+10%}) with neutral line or
Directly to two-phase supply network (120 V_{-10%} to 240 V_{+10%}) without neutral line
- Supply networks
- Leave L3 open circuit
- Filtering to be provided by the user.
- Fusing (such as fusible cut-outs) to be provided by the user

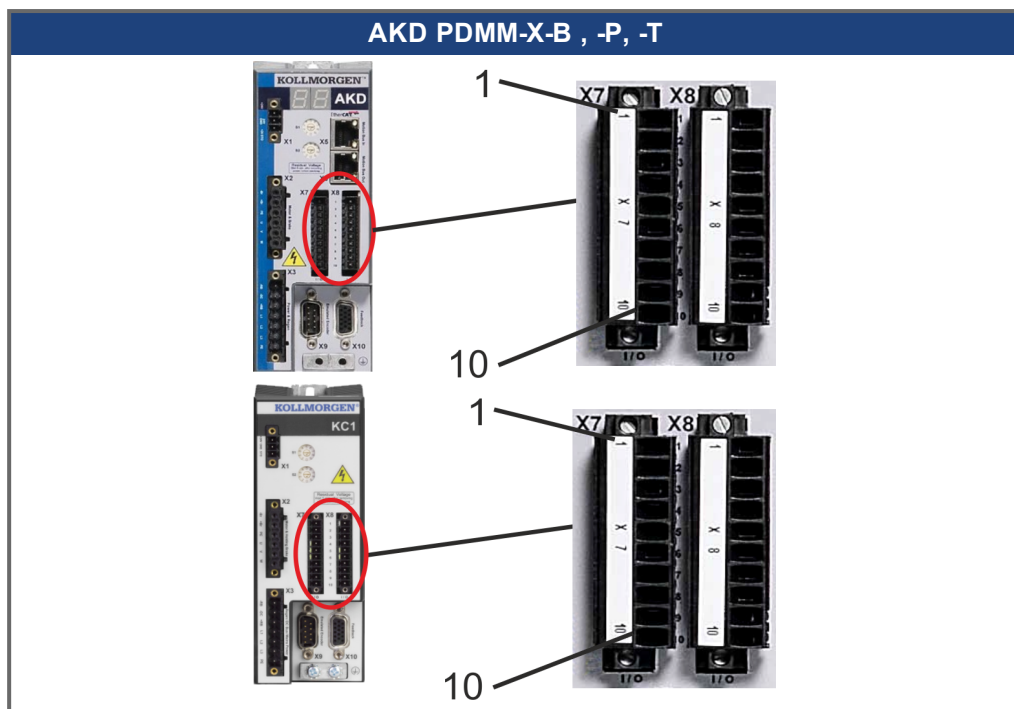


17.9 I/O Connection

17.9.1 Overview

17.9.1.1 I/O connectors X7 and X8 (all AKD PDMM variants)

Standard digital and analog I/O signals are connected to X7 and X8.



| Conn. | Pin | Signal | Abbreviation | Function |
|-------|-----|--------------------|--------------------|--|
| X7 | 1 | Digital Common X7 | DCOM7 | Common line for X7 pins 2, 3, 4, 9, 10 |
| X7 | 2 | Digital Input 7 | DIGITAL-IN 7 | Programmable |
| X7 | 3 | Digital Input 4 | DIGITAL-IN 4 | Programmable |
| X7 | 4 | Digital Input 3 | DIGITAL-IN 3 | Programmable |
| X7 | 5 | Digital Output 2- | DIGITAL-OUT2- | Programmable |
| X7 | 6 | Digital Output 2+ | DIGITAL-OUT2+ | Programmable |
| X7 | 7 | Digital Output 1- | DIGITAL-OUT1- | Programmable |
| X7 | 8 | Digital Output 1+ | DIGITAL-OUT1+ | Programmable |
| X7 | 9 | Digital Input 2 | DIGITAL-IN 2 | Programmable,fast |
| X7 | 10 | Digital Input 1 | DIGITAL-IN 1 | Programmable,fast |
| X8 | 1 | Fault Relay Output | Fault Relay Output | Fault Relay Output |
| X8 | 2 | Fault Relay Output | Fault Relay Output | Fault Relay Output |
| X8 | 3 | Digital Common X8 | DCOM8 | Common line for X8 pins 4, 5, 6 |
| X8 | 4 | Digital Input 8 | DIGITAL-IN 8 | Output stage enable, not programmable |
| X8 | 5 | Digital Input 6 | DIGITAL-IN 6 | Programmable |
| X8 | 6 | Digital Input 5 | DIGITAL-IN 5 | Programmable |
| X8 | 7 | Analog Ground | AGND | Analog GND |
| X8 | 8 | Analog Output + | Analog-Out | Actual velocity voltage |

| Conn. | Pin | Signal | Abbreviation | Function |
|-------|-----|----------------|--------------|--------------------|
| X8 | 9 | Analog Input - | Analog-In- | Velocity set point |
| X8 | 10 | Analog Input + | Analog-In+ | |

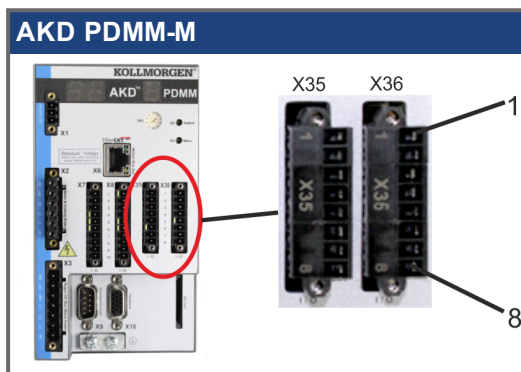
Digital common lines for X7 and X8 are not common to each other.

The DCOMx line should be connected to the 0V of the I/O supply when using sensors of type "Source" with digital inputs.

The DCOMx line should be connected to the 24V of the I/O supply when using sensors of type "Sink" with digital inputs.

17.9.1.2 I/O connectors X35 and X36 (AKD PDMM only)

AKD PDMM offers two additional connectors X35 and X36 for digital I/O signals.



| Conn. | Pin | Signal | Abbreviation | Function |
|-------|-----|--------------------|----------------|----------------------------------|
| X35 | 1 | Digital Common X35 | DCOM35 | Common line for X35 pins 2, 3, 4 |
| X35 | 2 | Digital Input 21 | DIGITAL-IN 21 | Programmable |
| X35 | 3 | Digital Input 22 | DIGITAL-IN 22 | Programmable |
| X35 | 4 | Digital Input 23 | DIGITAL-IN 23 | Programmable |
| X35 | 5 | n.c. | n.c. | - |
| X35 | 6 | n.c. | n.c. | - |
| X35 | 7 | Digital Output 21- | DIGITAL-OUT21- | Programmable |
| X35 | 8 | Digital Output 21+ | DIGITAL-OUT21+ | Programmable |
| X36 | 1 | Digital Common X36 | DCOM36 | Common line for X36 pins 2, 3, 4 |
| X36 | 2 | Digital Input 24 | DIGITAL-IN 24 | Programmable |
| X36 | 3 | Digital Input 25 | DIGITAL-IN 25 | Programmable |
| X36 | 4 | Digital Input 26 | DIGITAL-IN 26 | Programmable |
| X36 | 5 | n.c. | n.c. | - |
| X36 | 6 | n.c. | n.c. | - |
| X36 | 7 | Digital Output 22- | DIGITAL-OUT22- | Programmable |
| X36 | 8 | Digital Output 22+ | DIGITAL-OUT22+ | Programmable |

Digital common lines for X35 and X36 are not common to each other.

The DCOMx line should be connected to the 0V of the I/O supply when using sensors of type "Source" with digital inputs.

The DCOMx line should be connected to the 24V of the I/O supply when using sensors of type "Sink" with digital inputs.

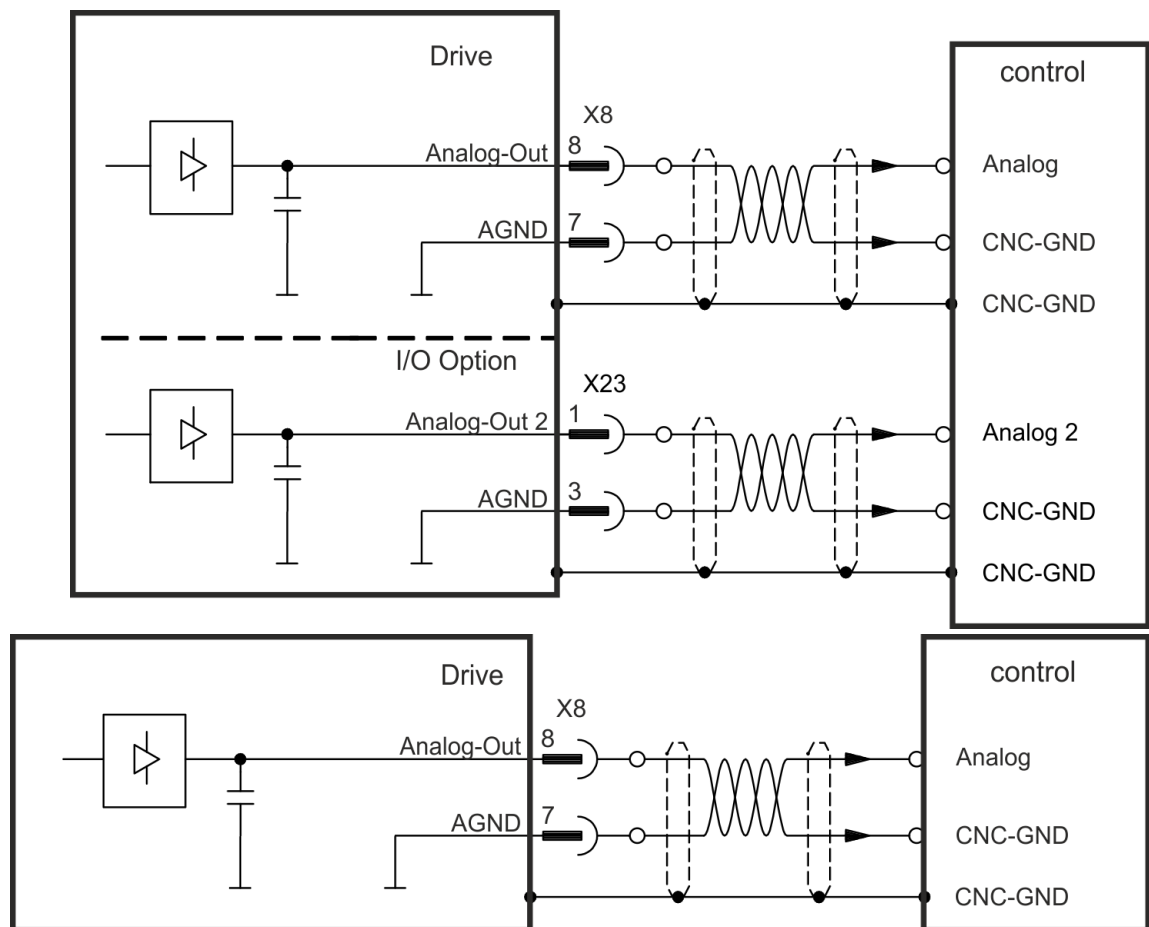
17.10 Analog Output (X8)

Analog Outputs can be used to output converted analog values of digital measurements recorded in the drive. The standard drive offers one analog output on X8. A list of the pre-programmed functions is included in the [EWW¹](#) setup software.

Technical characteristics

- Output voltage range referring to AGND: ± 10 V
- Resolution: 16 Bit and fully monotonic
- Update rate: 4 kHz
- Unadjusted offset: < 50 mV
- Offset drift typ: $250 \mu\text{V}/^\circ\text{C}$
- Gain or slope tolerance: $\pm 3\%$
- Nonlinearity: $< 0.1\%$ of full scale or 20 mV
- Output impedance: 110 ohms
- Specification complies with IEC 61131-2 Table 11
- -3 dB Bandwidth: > 8 kHz
- Maximum output current: 20 mA
- Capacitive load: any value but response speed limited by max Iout and by Rout
- Protected for short circuit to AGND

Analog Output Wiring Diagram



¹Embedded Workbench Views

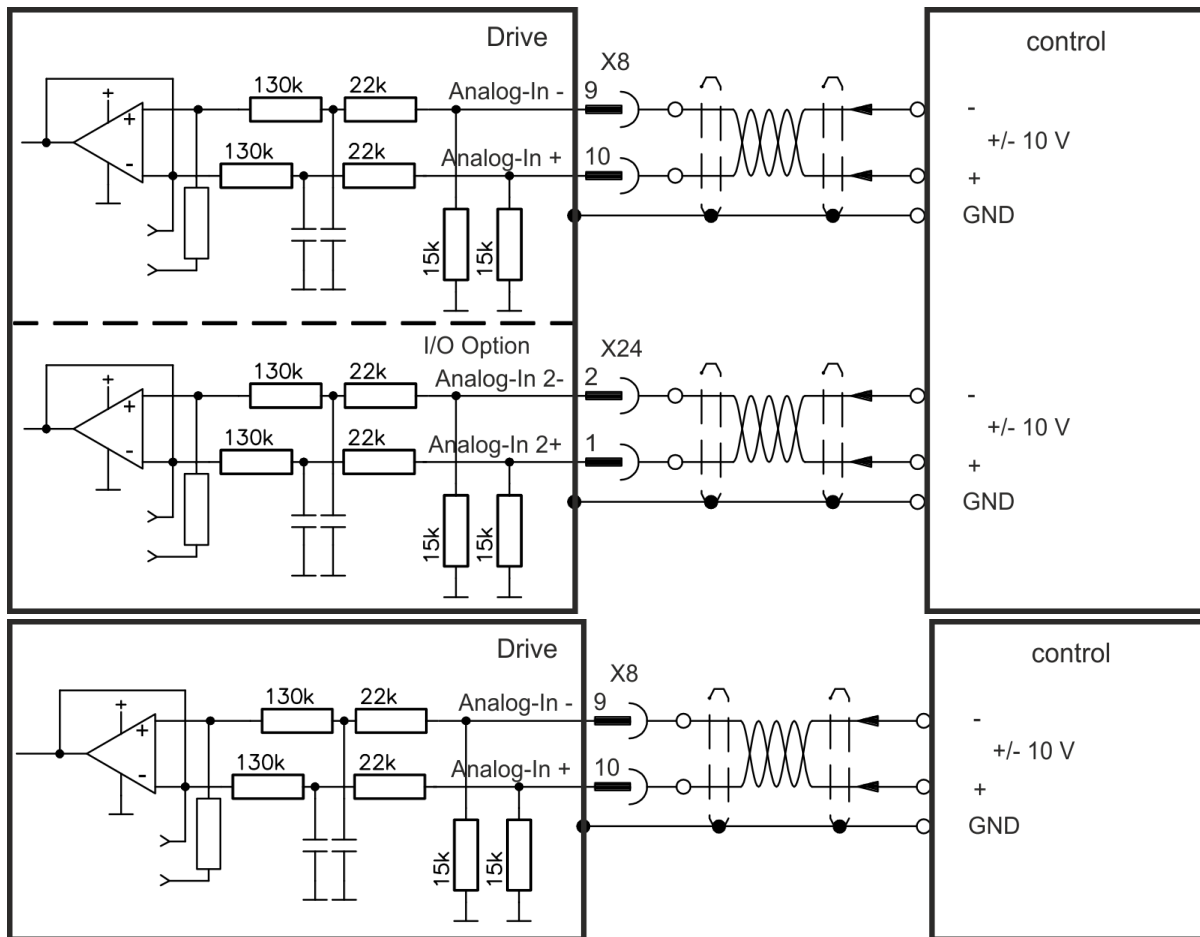
17.11 Analog Input (X8)

The drive is fitted with differential inputs for analog torque, velocity, or position control. The standard drive offers one analog input on X8.

Technical characteristics

- Differential input voltage range: $\pm 12.5\text{ V}$
- Maximum input voltage referring to I/O Return: $-12.5, +16.0\text{ V}$
- Resolution: 16 Bit and fully monotonic
- Firmware update rate: 16 kHz
- Unadjusted offset: $< 50\text{ mV}$
- Offset drift typ: $250\ \mu\text{V} / ^\circ\text{C}$
- Gain or slope tolerance: $\pm 3\%$
- Nonlinearity: $< 0.1\%$ of full scale or 12.5 mV
- Common Mode Rejection Ratio: $> 30\text{ dB}$ at 60 Hz
- Input impedance: $> 13\text{ k Ohms}$
- Signal to noise ratio referred to full scale:
 - AIN.CUTOFF = 3000 Hz: 14 bit
 - AIN.CUTOFF = 800 Hz: 16 bit

Analog Input Wiring Diagram



Application examples for set point input Analog-In:

- reduced-sensitivity input for setting-up/jog operation
- pre-control/override

Defining the direction of rotation

Standard setting: clockwise rotation of the motor shaft (looking at the shaft end) affected by positive voltage between terminal (+) and terminal (-)

To reverse the direction of rotation, swap the connections to terminals +/-, or change the DRV.DIR parameter in the "Feedback 1" screen page.

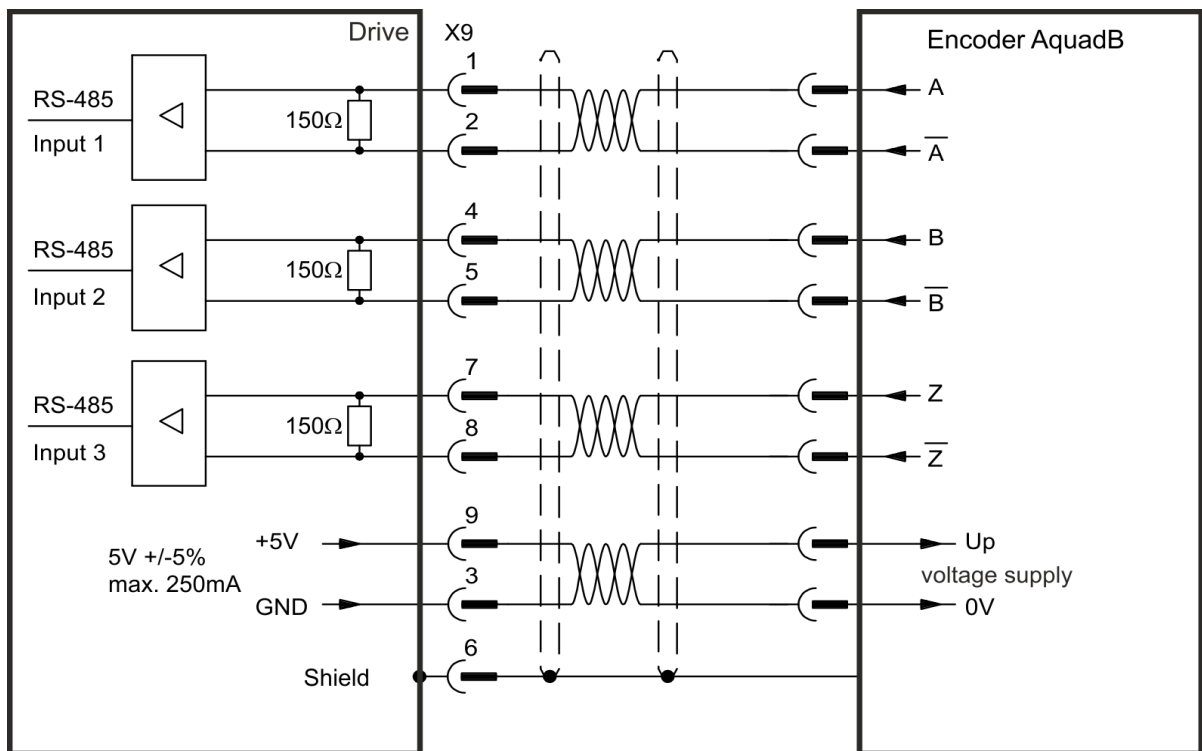
17.12 Command encoder signal connection

17.12.1 Incremental encoder input 5 V (X9)

A 5 V A quad B encoder, or the encoder emulation output of another drive can be connected to this input and used as a commander encoder, dual loop feedback, gearing or camming input. Parameter setting FB2.MODE = 0, FB2.SOURCE=1.

Don't use for primary motor feedback connection!

Connection Diagram

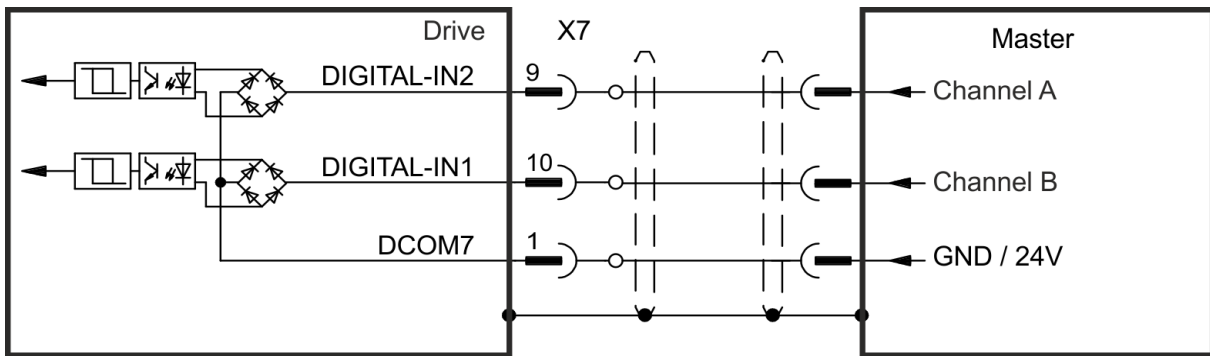


17.12.2 Incremental encoder input 24 V (X7)

A 24 V A quad B encoder can be connected to the digital inputs 1 and 2 and used as a commander encoder, dual loop feedback, gearing or camming input. Parameter setting FB2.MODE = 0, FB2.SOURCE=2.

Don't use for primary motor feedback connection!

Connection Diagram



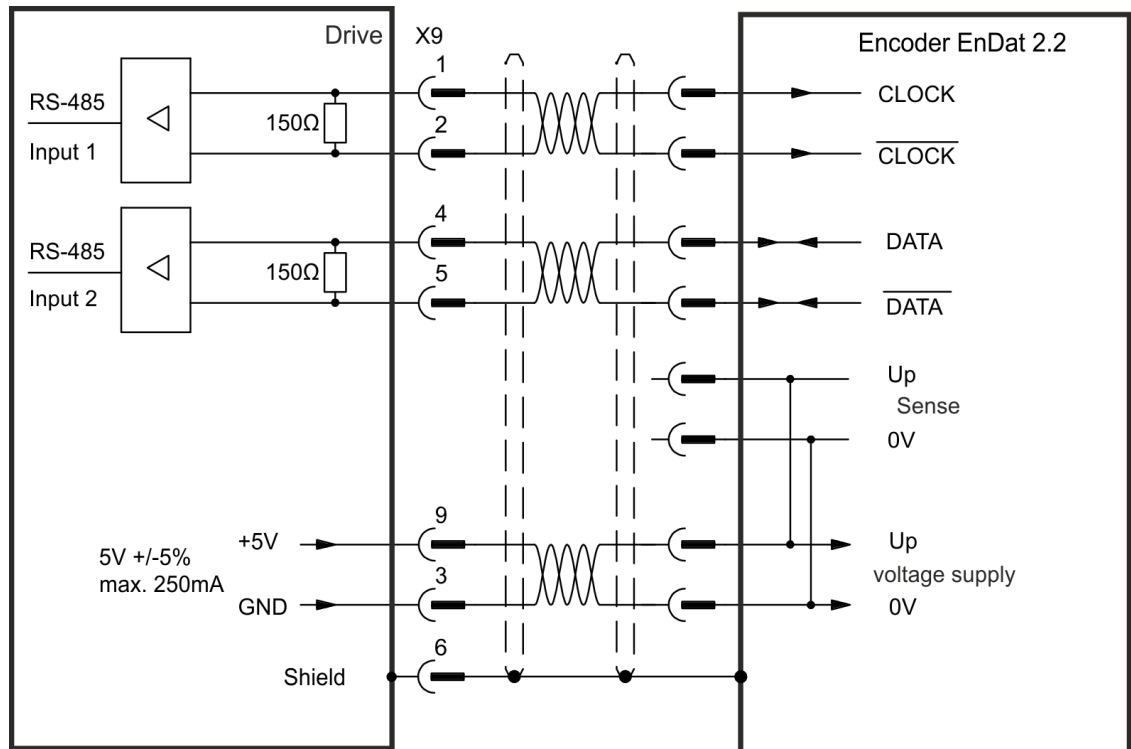
17.12.3 Encoder with EnDat 2.2 input 5 V (X9)

A single-turn or multi-turn encoder with EnDat 2.2 can be connected to this input and used as a commander encoder, dual loop feedback, gearing or camming input.

Parameter setting FB3.MODE=0, DRV.EMUEMODE=11.

Can be used as primary motor feedback connectio!

Connection Diagram

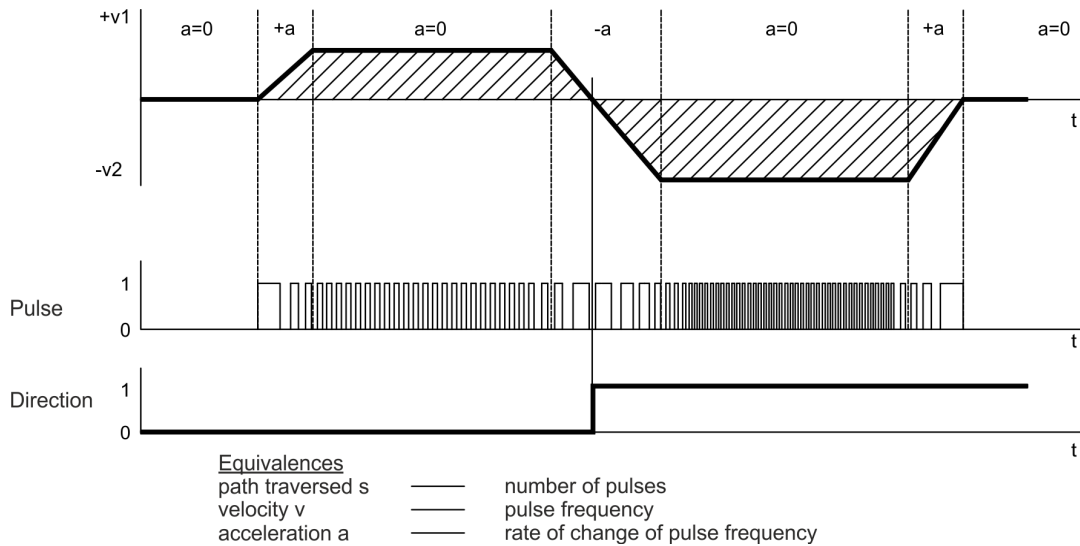


17.13 Pulse / Direction signal connection

The drive can be connected to a stepper-motor controller. Set parameters for the drive with EWW¹. The number of pulses can be adjusted, so that the drive can be adapted to match any stepper controller.

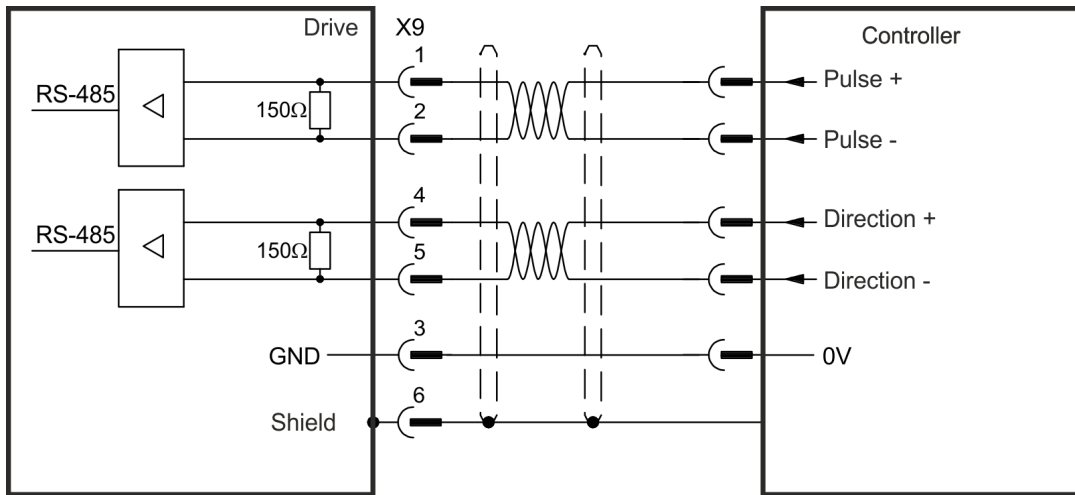
Speed profile and signal diagram

¹Embedded Workbench Views



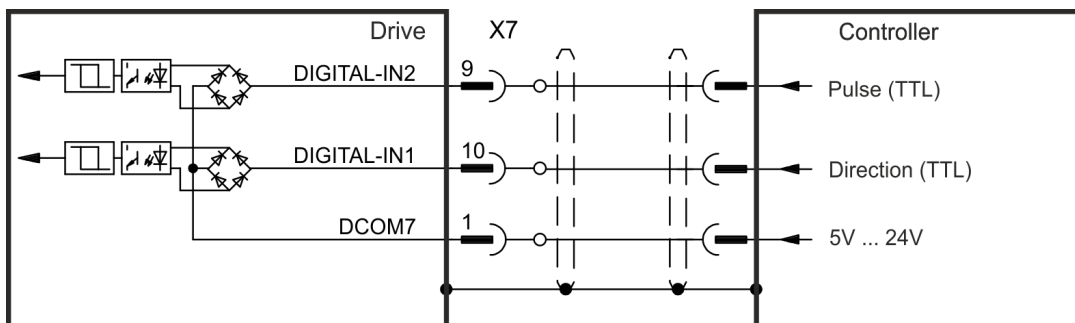
17.13.1 Pulse / Direction input 5 V (X9)

Connection to 5 V signal level stepper-motor controllers.



17.13.2 Pulse / Direction Input 5 V to 24 V (X7)

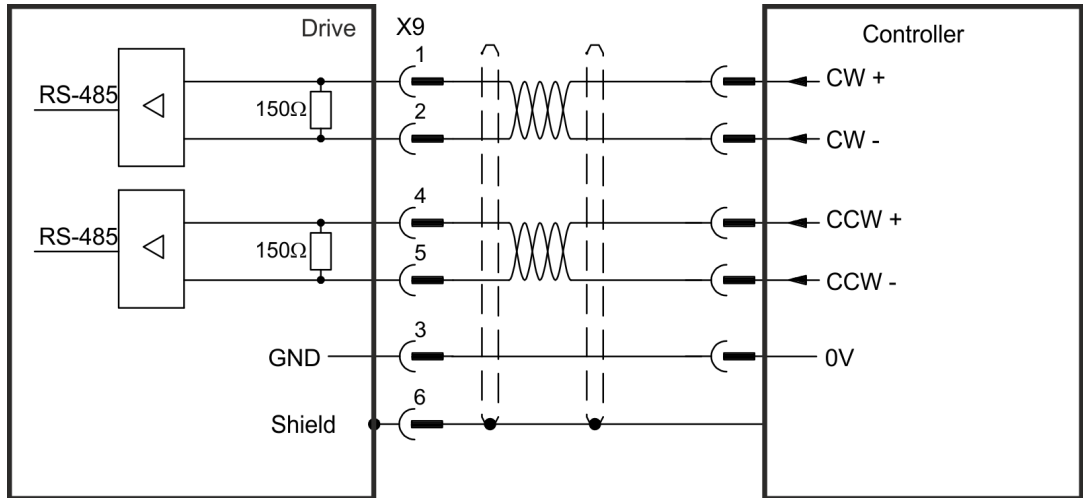
Connection industry standard 5V logic stepper-motor controllers with Pulse/Direction or Step/Direction outputs. Note that the X7 opto inputs can work with 5V up to 24V logic and so these inputs can be driven by 24V logic inputs as well.



17.14 CW / CCW signal connection

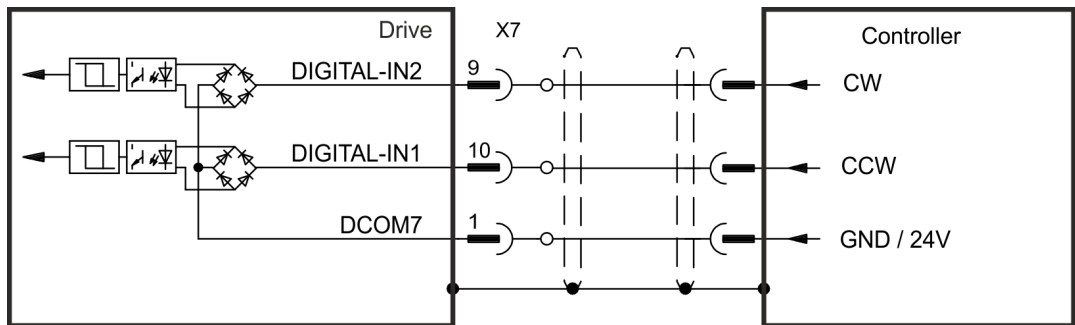
17.14.1 CW / CCW input 5 V (X9)

The drive can be connected to a third-party controller which delivers 5 V CW/CCW signals

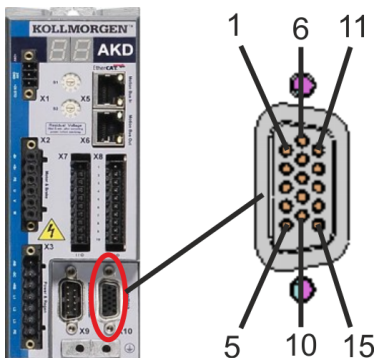


17.14.2 CW / CCW input 24 V (X7)

The drive can be connected to a third-party controller which delivers 24 V CW/CCW signals.



17.15 Feedback connector (X10)



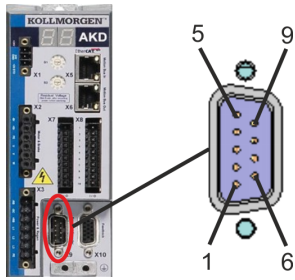
| Pin | SFD | SFD3/ DSL | Resolver | BiSS B (analog) | BiSS C (digital) | EnDAT 2.1 | EnDAT 2.2 | Hiper- face | Sine Enc. +Hall | Incr. Enc. +Hall | Hall | Tamagawa Smart Abs* |
|-----|-----|--------------|----------|--------------------|---------------------|--------------|--------------|----------------|-----------------------|------------------------|--------|---------------------------|
| 1 | - | - | - | - | - | - | - | - | Hall U | Hall U | Hall U | - |

| Pin | SFD | SFD3/ DSL | Resolver | BiSS B (analog) | BiSS C (digital) | EnDAT 2.1 | EnDAT 2.2 | Hiper- face | Sine Enc. +Hall | Incr. Enc. +Hall | Hall | Tamagawa Smart Abs* |
|-----|------|--------------|---------------------|--------------------|---------------------|--------------|--------------|----------------|-----------------------|------------------------|--------|---------------------------|
| 2 | - | - | - | CLK+ | CLK+ | CLK+ | CLK+ | - | Hall V | Hall V | Hall V | - |
| 3 | - | - | - | CLK- | CLK- | CLK- | CLK- | - | Hall W | Hall W | Hall W | - |
| 4 | SEN+ | - | - | SEN+ | SEN+ | SEN+ | SEN+ | SEN+ | SEN+ | SEN+ | - | SEN+ |
| 5 | SEN- | - | - | SEN- | SEN- | SEN- | SEN- | SEN- | SEN- | SEN- | - | SEN- |
| 6 | COM+ | COM+ | R1 Ref+ | DAT+ | DAT+ | DAT+ | DAT+ | DAT+ | Zero+ | Zero+ | - | SD+ |
| 7 | COM- | COM- | R2 Ref- | DAT- | DAT- | DAT- | DAT- | DAT- | Zero- | Zero- | - | SD- |
| 8 | - | - | Thermal control (+) | | | | | | | | | |
| 9 | - | - | Thermal control (-) | | | | | | | | | |
| 10 | +5 V | +5 V | - | +5 V | +5 V | +5 V | +5 V | +8 to +9 V | +5 V | +5 V | +5 V | +5 V |
| 11 | 0 V | 0 V | - | 0 V | 0 V | 0 V | 0 V | 0 V | 0 V | 0 V | 0 V | 0 V |
| 12 | - | - | S1 SIN+ | A+ | - | A+ | - | SIN+ | A+ | A+ | - | - |
| 13 | - | - | S3 SIN- | A- | - | A- | - | SIN- | A- | A- | - | - |
| 14 | - | - | S2 COS+ | B+ | - | B+ | - | COS+ | B+ | B+ | - | - |
| 15 | - | - | S4 COS- | B- | - | B- | - | COS- | B- | B- | - | - |

CLK = CLOCK, DAT = DATA, SEN = SENSE, *= for AKD PDMM with "NB" (rev 8+) only

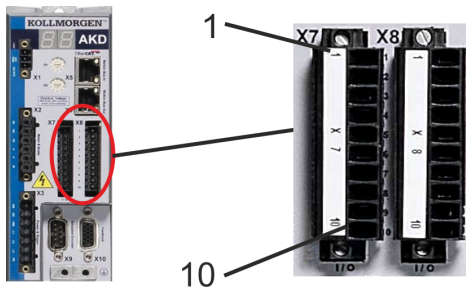
| Pin | SFD | SFD3/ DSL | Incr. Enc. +Hall | Hall |
|-----|--------|--------------|---------------------|--------|
| 1 | - | - | Hall U | Hall U |
| 2 | - | - | Hall V | Hall V |
| 3 | - | - | Hall W | Hall W |
| 4 | SENSE+ | - | SENSE+ | - |
| 5 | SENSE- | - | SENSE- | - |
| 6 | COM+ | COM+ | Zero+ | - |
| 7 | COM- | COM- | Zero- | - |
| 8 | - | - | Thermal control (+) | |
| 9 | - | - | Thermal control (-) | |
| 10 | +5 V | +5 V | +5 V | +5 V |
| 11 | 0 V | 0 V | 0 V | 0 V |
| 12 | - | - | A+ | - |
| 13 | - | - | A- | - |
| 14 | - | - | B+ | - |
| 15 | - | - | B- | - |

17.16 Feedback connector (X9)



| Pin | Pulse/Direction | CW/CCW | Incremental Encoder | Encoder with EnDat 2.2 |
|-----|-----------------|--------|------------------------|------------------------|
| 1 | Pulse+ | CW+ | A+ | CLOCK+ |
| 2 | Pulse- | CW- | A- | CLOCK- |
| 3 | GND | GND | GND | GND |
| 4 | Direction+ | CCW+ | B+ | DATA+ |
| 5 | Direction- | CCW- | B- | DATA- |
| 6 | Shield | Shield | Shield | Shield |
| 7 | - | - | Zero+ | - |
| 8 | - | - | Zero- | - |
| 9 | - | - | + 5 V (supply, output) | +5V (supply, output) |

17.17 Feedback connector (X7)



| Pin | Pulse/Direction | CW/CCW | Incremental Encoder |
|-----|-----------------|--------|---------------------|
| 9 | Pulse | CW | Channel A |
| 10 | Direction | CCW | Channel B |
| 1 | Common | Common | Common |

18 AKD PDMM Firmware

18.1 Check AKD PDMM Drive Firmware

To check AKD PDMM Firmware with KAS IDE:

1. Create a new project
2. Set the controller type and IP address
3. In the Project Explorer, double-click the **EtherCAT** node to open the EtherCAT Devices summary window
4. In the Devices tab, click the **Scan Devices** button
5. Choose the **Create...** option to map the physical device to a new device
6. If the version is not correct, continue following with the procedure below
7. Compile the project
8. Connect to the controller
9. Download the project to the controller
10. If the version is not correct, download the new firmware ([click here](#)) and follow with the next procedure.

18.2 Download AKD PDMM Drive Firmware

| | |
|---|----------|
| Component | Download |
| AKD PDMM Drive FW - Version: AKD-M-MCEC-V01-06-00-003.i00 | 4 MB |

To upgrade AKD PDMM Firmware with KAS IDE, continue from the previous procedure as follows:

1. In the EtherCAT Devices summary window, deactivate the **Online Configuration Mode**
2. Select the drives to be updated with the same firmware
(Note that the FW download is limited to 16 drives at a time)
3. Click **Upgrade Firmware** button

⚠ WARNING

Give careful attention to any warning dialog that appears at this point. These warnings usually include important information about preventing damage to the drives.

3. Browse to select the new AKD firmware file
4. Click **Open** to start the updating procedure

18.3 AKD PDMM Firmware Update

Based on the FoE protocol, the AKD PDMM Firmware can be downloaded as follows:

1. Scan the devices and make sure all devices are created
2. Compile the project
3. Connect to the controller
4. Download the project to the controller
5. Open the **EtherCAT Devices** summary form
6. Select the drives to be updated with the same firmware
7. Click the Upgrade Firmware button

⚠ WARNING

Give careful attention to any warning dialog that appears at this point. These warnings usually include important information about preventing damage to the drives.

8. Browse to select the new AKD firmware file
9. Click **Open** to start the updating procedure

NOTE

This procedure is not possible when applications are running and when the drive is in Online Configuration Mode.

During the firmware download, the AKD Firmware Update window displays a progress bar and the following messages are displayed:

- Uploading firmware to the drive
During the download process, the drive LED displays [dL]. Additional codes may appear during the download; see 3.2 Display Codes for a description of codes related to the firmware download.
- Resetting the drive
- Firmware update is complete

⚠ CAUTION

While the firmware is downloading to your drive, do not remove the 24V logic power. If you remove the 24V logic power during a firmware download, a severe drive crash can occur. If a crash occurs, the drive will restart in a special mode and prompt you to reload the firmware.

19 Parameter and Command Reference Guide

| | |
|---|------------|
| 19.1 About the Parameter and Command Reference Guide | 249 |
| 19.1.1 Parameter and Command Naming Conventions | 251 |
| 19.1.2 Summary of Parameters and Commands | 252 |
| 19.2 AOUT Parameters | 269 |
| 19.2.1 AOUT.CUTOFF | 270 |
| 19.2.2 AOUT.DEBUGADDR | 271 |
| 19.2.3 AOUT.DEBUGDATATYPE | 272 |
| 19.2.4 AOUT.DEBUGSCALE | 273 |
| 19.2.5 AOUT.ISCALE | 274 |
| 19.2.6 AOUT.MODE | 275 |
| 19.2.7 AOUT.OFFSET | 276 |
| 19.2.8 AOUT.PSCALE | 277 |
| 19.2.9 AOUT.VALUE | 278 |
| 19.2.10 AOUT.VALUEU | 279 |
| 19.2.11 AOUT.VSCALE | 280 |
| 19.3 BODE Parameters | 281 |
| 19.3.1 BODE.EXCITEGAP | 282 |
| 19.3.2 BODE.FREQ | 283 |
| 19.3.3 BODE.IAMP | 284 |
| 19.3.4 BODE.IFLIMIT | 285 |
| 19.3.5 BODE.IFTHRESH | 286 |
| 19.3.6 BODE.INJECTPOINT | 287 |
| 19.3.7 BODE.MODE | 288 |
| 19.3.8 BODE.MODETIMER | 291 |
| 19.3.9 BODE.PRBDEPTH | 293 |
| 19.3.10 BODE.VAMP | 294 |
| 19.3.11 BODE.VFLIMIT | 295 |
| 19.3.12 BODE.VFTHRESH | 296 |
| 19.4 CMP Parameters | 297 |
| 19.4.1 CMPx.ADVANCE | 298 |
| 19.4.2 CMPx.ARM | 299 |
| 19.4.3 CMPx.MODBOUND1 | 300 |

| | |
|---|------------|
| 19.4.4 CMPx.MODBOUND2 | 302 |
| 19.4.5 CMPx.MODE | 304 |
| 19.4.6 CMPx.MODEN | 306 |
| 19.4.7 CMPx.MODVALUE | 307 |
| 19.4.8 CMPx.OUTMASK | 308 |
| 19.4.9 CMPx.SETPOINT | 309 |
| 19.4.10 CMPx.SOURCE | 311 |
| 19.4.11 CMPx.SOURCEVALUE | 312 |
| 19.4.12 CMPx.STATE | 313 |
| 19.4.13 CMPx.WIDTH | 314 |
| 19.4.14 CMPx.WIDTHTYPE | 316 |
| 19.5 COGCOMP Parameters | 318 |
| 19.5.1 COGCOMP.CORRECTIONTABLE | 319 |
| 19.5.2 COGCOMP.CORRECTIONVALUE | 320 |
| 19.5.3 COGCOMP.EN | 321 |
| 19.5.4 COGCOMP.LOAD | 322 |
| 19.5.5 COGCOMP.RANGEHIGH | 323 |
| 19.5.6 COGCOMP.RANGELOW | 324 |
| 19.5.7 COGCOMP.SAVE | 325 |
| 19.5.8 COGCOMP.TEACH | 326 |
| 19.5.9 COGCOMP.V | 327 |
| 19.6 CS Parameters | 328 |
| 19.6.1 CS.DEC | 329 |
| 19.6.2 CS.STATE | 330 |
| 19.6.3 CS.TO | 331 |
| 19.6.4 CS.VTHRESH | 332 |
| 19.7 DIN Parameters | 333 |
| 19.7.1 DIN.HCMD1 TO DIN.HCMD4 | 334 |
| 19.7.2 DIN.LCMD1 to DIN.LCMD4 | 335 |
| 19.7.3 DIN.ROTARY | 336 |
| 19.7.4 DIN.STATES | 337 |
| 19.7.5 DIN1.FILTER TO DIN7.FILTER | 338 |
| 19.7.6 DIN1.INV to DIN7.INV | 339 |
| 19.7.7 DIN1.MODE TO DIN7.MODE | 340 |

| | |
|---|------------|
| 19.7.8 DIN1.PARAM TO DIN7.PARAM | 342 |
| 19.7.9 DIN1.STATE TO DIN7.STATE | 344 |
| 19.7.10 DIN9.STATE to DIN11.STATE | 345 |
| 19.8 DIO Parameters | 346 |
| 19.8.1 DIO9.INV to DIO11.INV | 347 |
| 19.8.2 DIO9.DIR to DIO11.DIR | 348 |
| 19.9 DOUT Parameters | 349 |
| 19.9.1 DOUT.CTRL | 350 |
| 19.9.2 DOUT.RELAYMODE | 351 |
| 19.9.3 DOUT.STATES | 352 |
| 19.9.4 DOUT1.MODE and DOUT2.MODE | 353 |
| 19.9.5 DOUT1.PARAM AND DOUT2.PARAM | 355 |
| 19.9.6 DOUT1.STATE AND DOUT2.STATE | 357 |
| 19.9.7 DOUT1.STATEU AND DOUT2.STATEU | 358 |
| 19.9.8 DOUT9.MODE to DOUT11.MODE | 359 |
| 19.9.9 DOUT9.PARAM to DOUT11.MODE | 360 |
| 19.9.10 DOUT9.STATE to DOUT11.STATE | 361 |
| 19.9.11 DOUT9.STATEU to DOUT11.STATEU | 362 |
| 19.10 DRV Parameters | 363 |
| 19.10.1 DRV.ACC | 366 |
| 19.10.2 DRV.ACTIVE | 368 |
| 19.10.3 DRV.BLINKDISPLAY | 369 |
| 19.10.4 DRV.BOOTTIME | 370 |
| 19.10.5 DRV.CLRCRASHDUMP | 371 |
| 19.10.6 DRV.CLRFAULTHIST | 372 |
| 19.10.7 DRV.CLRFAULTS | 373 |
| 19.10.8 DRV.CMDDELAY | 374 |
| 19.10.9 DRV.CMDSOURCE | 375 |
| 19.10.10 DRV.CRASHDUMP | 376 |
| 19.10.11 DRV.CUSTOMIDENTIFIER | 377 |
| 19.10.12 DRV.DBILIMIT | 378 |
| 19.10.13 DRV.DEC | 379 |
| 19.10.14 DRV.DIFVAR | 381 |
| 19.10.15 DRV.DIR | 382 |

| | |
|------------------------------------|-----|
| 19.10.16 DRV.DIS | 384 |
| 19.10.17 DRV.DISMODE | 385 |
| 19.10.18 DRV.DISSOURCES | 386 |
| 19.10.19 DRV.DISSOURCESMASK | 387 |
| 19.10.20 DRV.DISTO | 388 |
| 19.10.21 DRV.DOWNLOADALLOWED | 389 |
| 19.10.22 DRV.EMUJCHECKSPEED | 390 |
| 19.10.23 DRV.EMUJDIR | 391 |
| 19.10.24 DRV.EMUJMODE | 392 |
| 19.10.25 DRV.EMUJTURN | 394 |
| 19.10.26 DRV.EMUJULSEWIDTH | 395 |
| 19.10.27 DRV.EMUJRES | 396 |
| 19.10.28 DRV.EMUJSTEPCMD | 397 |
| 19.10.29 DRV.EMUJSTEPMDPIN | 398 |
| 19.10.30 DRV.EMUJSTEPMDPOUT | 399 |
| 19.10.31 DRV.EMUJZOFFSET | 400 |
| 19.10.32 DRV.EN | 401 |
| 19.10.33 DRV.ENDEFAULT | 402 |
| 19.10.34 DRV.ERRORLIST | 403 |
| 19.10.35 DRV.FAULTDISPLAYMODE | 404 |
| 19.10.36 DRV.FAULTHIST | 405 |
| 19.10.37 DRV.FAULT1 to DRV.FAULT10 | 406 |
| 19.10.38 DRV.FAULTS | 407 |
| 19.10.39 DRV.HANDWHEELSRC | 408 |
| 19.10.40 DRV.HELP | 409 |
| 19.10.41 DRV.HELPALL | 410 |
| 19.10.42 DRV.HWENABLE | 411 |
| 19.10.43 DRV.HWENDELAY | 412 |
| 19.10.44 DRV.HWENMODE | 413 |
| 19.10.45 DRV.ICONT | 414 |
| 19.10.46 DRV.INFO | 415 |
| 19.10.47 DRV.IPEAK | 417 |
| 19.10.48 DRV.IZERO | 418 |
| 19.10.49 DRV.LIST | 419 |

| | |
|---|------------|
| 19.10.50 DRV.LOGICVOLTS | 420 |
| 19.10.51 DRV.MEMADDR | 421 |
| 19.10.52 DRV.MEMDATA | 422 |
| 19.10.53 DRV.MOTIONDISSOURCES | 423 |
| 19.10.54 DRV.NAME | 424 |
| 19.10.55 DRV.NVCHECK | 425 |
| 19.10.56 DRV.NVLIST | 426 |
| 19.10.57 DRV.NVLOAD | 427 |
| 19.10.58 DRV.NVSAVE | 428 |
| 19.10.59 DRV.ONTIME | 429 |
| 19.10.60 DRV.OPMODE | 430 |
| 19.10.61 DRV.READFORMAT | 431 |
| 19.10.62 DRV.REBOOT | 432 |
| 19.10.63 DRV.RSTVAR | 433 |
| 19.10.64 DRV.RUNTIME | 434 |
| 19.10.65 DRV.SETUPREQBITS | 435 |
| 19.10.66 DRV.SETUPREQLIST | 437 |
| 19.10.67 DRV.STOP | 438 |
| 19.10.68 DRV.TEMPERATURES | 439 |
| 19.10.69 DRV.TYPE | 440 |
| 19.10.70 DRV.VER | 441 |
| 19.10.71 DRV.VERIMAGE | 442 |
| 19.10.72 DRV.WARNING1 to DRV.WARNING3 | 443 |
| 19.10.73 DRV.WARNINGS | 444 |
| 19.10.74 DRV.ZERO | 445 |
| 19.11 DS402 Parameters | 446 |
| 19.11.1 DS402.1ADDPOSFCFEED | 447 |
| 19.11.2 DS402.1ADDPOSFCSHAFTREV | 448 |
| 19.11.3 DS402.3ADDPOSFCFEED | 449 |
| 19.11.4 DS402.3ADDPOSFCSHAFTREV | 450 |
| 19.12 FAULT Parameters | 451 |
| 19.12.1 FAULTx.ACTION | 452 |
| 19.12.2 FAULT570.ACTION | 453 |
| 19.13 FB1 Parameters | 454 |

| | |
|-------------------------------------|-----|
| 19.13.1 FB1.BISSBITS | 456 |
| 19.13.2 FB1.CALTHRESH | 457 |
| 19.13.3 FB1.DIAG | 458 |
| 19.13.4 FB1.ENCRES | 459 |
| 19.13.5 FB1.EXTENDEDMULTITURN | 460 |
| 19.13.6 FB1.FAULTS | 461 |
| 19.13.7 FB1.HALLSTATE | 463 |
| 19.13.8 FB1.HALLSTATEU | 464 |
| 19.13.9 FB1.HALLSTATEV | 465 |
| 19.13.10 FB1.HALLSTATEW | 466 |
| 19.13.11 FB1.IDENTIFIED | 467 |
| 19.13.12 FB1.INITPSAVED | 468 |
| 19.13.13 FB1.INITPSTATUS | 469 |
| 19.13.14 FB1.INITPWINDOW | 470 |
| 19.13.15 FB1.INITSIGNED | 471 |
| 19.13.16 FB1.LASTIDENTIFIED | 472 |
| 19.13.17 FB1.MECHPOS | 473 |
| 19.13.18 FB1.MEMVER | 474 |
| 19.13.19 FB1.MOTORPHASE | 475 |
| 19.13.20 FB1.MOTORPOLES | 476 |
| 19.13.21 FB1.OFFSET | 477 |
| 19.13.22 FB1.ORIGIN | 478 |
| 19.13.23 FB1.P | 480 |
| 19.13.24 FB1.PDIR | 481 |
| 19.13.25 FB1.PFIND | 482 |
| 19.13.26 FB1.PFINDCMDU | 483 |
| 19.13.27 FB1.PIN | 484 |
| 19.13.28 FB1.POFFSET | 485 |
| 19.13.29 FB1.POLES | 486 |
| 19.13.30 FB1.POUT | 487 |
| 19.13.31 FB1.PSCALE | 488 |
| 19.13.32 FB1.PUNIT | 489 |
| 19.13.33 FB1.RESKTR | 490 |
| 19.13.34 FB1.RESREFPHASE | 491 |

| | |
|---|------------|
| 19.13.35 FB1.SELECT | 492 |
| 19.13.36 FB1.TRACKINGCAL | 494 |
| 19.13.37 FB1.USERBYTE0 to FB1.USERBYTE7 | 495 |
| 19.13.38 FB1.USERDWORD0 to FB1.USERDWORD1 | 497 |
| 19.13.39 FB1.USERWORD0 to FB1.USERWORD3 | 499 |
| 19.14 FB3 Parameters | 501 |
| 19.14.1 FB3.DIR | 502 |
| 19.14.2 FB3.ENCREC | 503 |
| 19.14.3 FB3.MODE | 504 |
| 19.14.4 FB3.MOTORPHASE | 505 |
| 19.14.5 FB3.MOTORPOLES | 506 |
| 19.14.6 FB3.P | 507 |
| 19.14.7 FB3.PDIR | 508 |
| 19.14.8 FB3.PIN | 509 |
| 19.14.9 FB3.POFFSET | 510 |
| 19.14.10 FB3.POUT | 511 |
| 19.14.11 FB3.PUNIT | 512 |
| 19.15 FBUS Parameters | 513 |
| 19.15.1 FBUS.PARAM1 TO FBUS.PARAM10 | 514 |
| 19.15.2 FBUS.PLLSTATE | 517 |
| 19.15.3 FBUS.PLLTHRESH | 518 |
| 19.15.4 FBUS.PROTECTION | 519 |
| 19.15.5 FBUS.REMOTE | 521 |
| 19.15.6 FBUS.SAMPLEPERIOD | 522 |
| 19.15.7 FBUS.STATE | 523 |
| 19.15.8 FBUS.SYNCACQUIREWND | 524 |
| 19.15.9 FBUS.SYNCACT | 525 |
| 19.15.10 FBUS.SYNCDIST | 526 |
| 19.15.11 FBUS.SYNCLOCKWND | 527 |
| 19.15.12 FBUS.SYNCWND | 528 |
| 19.15.13 FBUS.TYPE | 529 |
| 19.16 GUI Parameters | 530 |
| 19.16.1 GUI.DISPLAY | 531 |
| 19.16.2 GUI.DISPLAYS | 532 |

| | |
|------------------------------------|------------|
| 19.16.3 GUI.PARAM01 | 533 |
| 19.16.4 GUI.PARAM02 | 534 |
| 19.16.5 GUI.PARAM03 | 535 |
| 19.16.6 GUI.PARAM04 | 536 |
| 19.16.7 GUI.PARAM05 | 537 |
| 19.16.8 GUI.PARAM06 | 538 |
| 19.16.9 GUI.PARAM07 | 539 |
| 19.16.10 GUI.PARAM08 | 540 |
| 19.16.11 GUI.PARAM09 | 541 |
| 19.16.12 GUI.PARAM10 | 542 |
| 19.17 HWLS Parameters | 543 |
| 19.17.1 HWLS.NEGSTATE | 544 |
| 19.17.2 HWLS.POSSTATE | 545 |
| 19.18 IL Parameters | 546 |
| 19.18.1 IL.BUSFF | 548 |
| 19.18.2 IL.CMD | 549 |
| 19.18.3 IL.CMDACC | 550 |
| 19.18.4 IL.CMDU | 551 |
| 19.18.5 IL.DCMD | 552 |
| 19.18.6 IL.DFB | 553 |
| 19.18.7 IL.DIFOLD | 554 |
| 19.18.8 IL.FB | 555 |
| 19.18.9 IL.FBSOURCE | 556 |
| 19.18.10 IL.FF | 557 |
| 19.18.11 IL.FOLDFTHRESH | 558 |
| 19.18.12 IL.FOLDFTHRESHU | 559 |
| 19.18.13 IL.FOLDWTHRESH | 560 |
| 19.18.14 IL.FRCTION | 561 |
| 19.18.15 IL.IFOLD | 562 |
| 19.18.16 IL.IUFB | 563 |
| 19.18.17 IL.IVFB | 564 |
| 19.18.18 IL.KACFF | 565 |
| 19.18.19 IL.KBUSFF | 566 |
| 19.18.20 IL.KP | 567 |

| | |
|-------------------------------------|------------|
| 19.18.21 IL.KPDRATIO | 568 |
| 19.18.22 IL.KPLOOKUPINDEX | 569 |
| 19.18.23 IL.KPLOOKUPVALUE | 570 |
| 19.18.24 IL.KPLOOKUPVALUES | 571 |
| 19.18.25 IL.KPSOURCE | 572 |
| 19.18.26 IL.KVFF | 573 |
| 19.18.27 IL.LIMITN | 574 |
| 19.18.28 IL.LIMITP | 575 |
| 19.18.29 IL.MFOLDD | 576 |
| 19.18.30 IL.MFOLDR | 577 |
| 19.18.31 IL.MFOLDT | 578 |
| 19.18.32 IL.MI2T | 579 |
| 19.18.33 IL.MI2TWTRESH | 580 |
| 19.18.34 IL.MIFOLD | 581 |
| 19.18.35 IL.MIMODE | 582 |
| 19.18.36 IL.OFFSET | 583 |
| 19.18.37 IL.VCMD | 584 |
| 19.18.38 IL.VLIMIT | 585 |
| 19.18.39 IL.VUFB | 586 |
| 19.18.40 IL.VVFB | 587 |
| 19.19 LOAD Parameters | 588 |
| 19.19.1 LOAD.INERTIA | 589 |
| 19.20 MOTOR Parameters | 590 |
| 19.20.1 MOTOR.AUTOSET | 592 |
| 19.20.2 MOTOR.BRAKE | 594 |
| 19.20.3 MOTOR.BRAKEIMM | 595 |
| 19.20.4 MOTOR.BRAKERLS | 596 |
| 19.20.5 MOTOR.BRAKESTATE | 597 |
| 19.20.6 MOTOR.CTF0 | 598 |
| 19.20.7 MOTOR.FIELDWEAKENING | 599 |
| 19.20.8 MOTOR.ICONT | 600 |
| 19.20.9 MOTOR.IDDATAVALID | 601 |
| 19.20.10 MOTOR.IDMAX | 602 |
| 19.20.11 MOTOR.IMID | 603 |

| | |
|--------------------------------------|------------|
| 19.20.12 MOTOR.IMTR | 604 |
| 19.20.13 MOTOR.INERTIA | 606 |
| 19.20.14 MOTOR.IPEAK | 607 |
| 19.20.15 MOTOR.KE | 608 |
| 19.20.16 MOTOR.KT | 609 |
| 19.20.17 MOTOR.LDLL | 610 |
| 19.20.18 MOTOR.LISAT | 611 |
| 19.20.19 MOTOR.LQLL | 612 |
| 19.20.20 MOTOR.NAME | 613 |
| 19.20.21 MOTOR.PHASE | 614 |
| 19.20.22 MOTOR.PHSADV1 | 615 |
| 19.20.23 MOTOR.PHSADV2 | 616 |
| 19.20.24 MOTOR.PITCH | 617 |
| 19.20.25 MOTOR.POLES | 618 |
| 19.20.26 MOTOR.R | 619 |
| 19.20.27 MOTOR.RTYPE | 620 |
| 19.20.28 MOTOR.SUPPORTEDPARAMS | 621 |
| 19.20.29 MOTOR.TBRAKEAPP | 622 |
| 19.20.30 MOTOR.TBRAKERLS | 623 |
| 19.20.31 MOTOR.TBRAKETO | 624 |
| 19.20.32 MOTOR.TEMP | 625 |
| 19.20.33 MOTOR.TEMPC | 626 |
| 19.20.34 MOTOR.TEMPFALT | 627 |
| 19.20.35 MOTOR.TEMPWARN | 628 |
| 19.20.36 MOTOR.TYPE | 629 |
| 19.20.37 MOTOR.VMAX | 631 |
| 19.20.38 MOTOR.VOLTMAX | 632 |
| 19.20.39 MOTOR.VOLTMIN | 633 |
| 19.20.40 MOTOR.VOLTRATED | 634 |
| 19.20.41 MOTOR.VRATED | 635 |
| 19.21 PL Parameters | 636 |
| 19.21.1 PL.CMD | 637 |
| 19.21.2 PL.ERR | 638 |
| 19.21.3 PL.ERRFTHRESH | 639 |

| | |
|-------------------------------------|------------|
| 19.21.4 PL.ERRMODE | 641 |
| 19.21.5 PL.ERRWTHRESH | 643 |
| 19.21.6 PL.FB | 644 |
| 19.21.7 PL.FBSOURCE | 645 |
| 19.21.8 PL.FILTERTIME | 646 |
| 19.21.9 PL.INTINMAX | 647 |
| 19.21.10 PL.INTOUTMAX | 649 |
| 19.21.11 PL.KI | 651 |
| 19.21.12 PL.KITHRESH | 652 |
| 19.21.13 PL.KP | 653 |
| 19.21.14 PL.PDELAY | 654 |
| 19.22 REC Parameters | 655 |
| 19.22.1 REC.ACTIVE | 656 |
| 19.22.2 REC.CH1 to REC.CH6 | 657 |
| 19.22.3 REC.DONE | 658 |
| 19.22.4 REC.GAP | 659 |
| 19.22.5 REC.NUMPOINTS | 660 |
| 19.22.6 REC.OFF | 661 |
| 19.22.7 REC.RECPRMLIST | 662 |
| 19.22.8 REC.RETRIEVE | 663 |
| 19.22.9 REC.RETRIEVEDATA | 664 |
| 19.22.10 REC.RETRIEVEFRMT | 666 |
| 19.22.11 REC.RETRIEVEHDR | 667 |
| 19.22.12 REC.RETRIEVESIZE | 668 |
| 19.22.13 REC.STOPTYPE | 669 |
| 19.22.14 REC.TRIG | 670 |
| 19.22.15 REC.TRIGPARAM | 671 |
| 19.22.16 REC.TRIGPOS | 672 |
| 19.22.17 REC.TRIGPRMLIST | 674 |
| 19.22.18 REC.TRIGSLOPE | 675 |
| 19.22.19 REC.TRIGTYPE | 676 |
| 19.22.20 REC.TRIGVAL | 677 |
| 19.23 REGEN Parameters | 678 |
| 19.23.1 REGEN.POWER | 679 |

| | |
|--|------------|
| 19.23.2 REGEN.POWERFILTERED | 680 |
| 19.23.3 REGEN.REXT | 681 |
| 19.23.4 REGEN.TEXT | 682 |
| 19.23.5 REGEN.TYPE | 683 |
| 19.23.6 REGEN.WATTEXT | 684 |
| 19.24 SM Parameters | 685 |
| 19.24.1 SM.ACC | 686 |
| 19.24.2 SM.DEC | 687 |
| 19.24.3 SM.I1 | 688 |
| 19.24.4 SM.I2 | 689 |
| 19.24.5 SM.MODE | 690 |
| 19.24.6 SM.MOVE | 693 |
| 19.24.7 SM.T1 | 694 |
| 19.24.8 SM.T2 | 695 |
| 19.24.9 SM.V1 | 696 |
| 19.24.10 SM.V2 | 697 |
| 19.25 STO Parameters | 698 |
| 19.25.1 STO.STATE | 699 |
| 19.26 SWLS Parameters | 700 |
| 19.26.1 SWLS.EN | 701 |
| 19.26.2 SWLS.LIMIT0 | 702 |
| 19.26.3 SWLS.LIMIT1 | 703 |
| 19.26.4 SWLS.STATE | 704 |
| 19.27 TEMP Parameters | 705 |
| 19.27.1 TEMP.CONTROL | 706 |
| 19.27.2 TEMP.POWER1 to TEMP.POWER3 | 707 |
| 19.28 USER Parameters | 708 |
| 19.28.1 USER.INT1 to USER.INT24 | 709 |
| 19.29 VBUS Parameters | 710 |
| 19.29.1 VBUS.HALFVOLT | 711 |
| 19.29.2 VBUS.OVFTHRESH | 712 |
| 19.29.3 VBUS.OVWTHRESH | 713 |
| 19.29.4 VBUS.RMSLIMIT | 714 |
| 19.29.5 VBUS.UVFTHRESH | 715 |

| | |
|--|------------|
| 19.29.6 VBUS.UVMODE | 716 |
| 19.29.7 VBUS.UVWTHRESH | 717 |
| 19.29.8 VBUS.VALUE | 718 |
| 19.30 VL Parameters | 719 |
| 19.30.1 VL.ARPF1 TO VL.ARPF4 | 720 |
| 19.30.2 VL.ARPQ1 TO VL.ARPQ4 | 722 |
| 19.30.3 VL.ARTYPE1 TO VL.ARTYPE4 | 724 |
| 19.30.4 VL.ARZF1 TO VL.ARZF4 | 725 |
| 19.30.5 VL.ARZQ1 TO VL.ARZQ4 | 726 |
| 19.30.6 VL.BUSFF | 728 |
| 19.30.7 VL.CMD | 729 |
| 19.30.8 VL.CMDU | 730 |
| 19.30.9 VL.ERR | 731 |
| 19.30.10 VL.FB | 732 |
| 19.30.11 VL.FBFILTER | 733 |
| 19.30.12 VL.FBSOURCE | 734 |
| 19.30.13 VL.FBUNFILTERED | 735 |
| 19.30.14 VL.FF | 736 |
| 19.30.15 VL.FFDELAY | 737 |
| 19.30.16 VL.GENMODE | 738 |
| 19.30.17 VL.KBUSFF | 739 |
| 19.30.18 VL.KI | 740 |
| 19.30.19 VL.KIMODE | 741 |
| 19.30.20 VL.KO | 742 |
| 19.30.21 VL.KP | 743 |
| 19.30.22 VL.KVFF | 745 |
| 19.30.23 VL.LIMITN | 746 |
| 19.30.24 VL.LIMITP | 748 |
| 19.30.25 VL.LMJR | 749 |
| 19.30.26 VL.MODEL | 750 |
| 19.30.27 VL.OBSBW | 751 |
| 19.30.28 VL.OBSMODE | 752 |
| 19.30.29 VL.THRESH | 753 |
| 19.30.30 VL.VFTHRESH | 755 |

| | |
|----------------------------------|------------|
| 19.31 WS Parameters | 756 |
| 19.31.1 WS.ARM | 757 |
| 19.31.2 WS.CHECKMODE | 758 |
| 19.31.3 WS.CHECKT | 759 |
| 19.31.4 WS.CHECKV | 760 |
| 19.31.5 WS.DISARM | 761 |
| 19.31.6 WS.DISTMAX | 762 |
| 19.31.7 WS.DISTMIN | 763 |
| 19.31.8 WS.FREQ | 764 |
| 19.31.9 WS.IMAX | 765 |
| 19.31.10 WS.MODE | 766 |
| 19.31.11 WS.NUMLOOPS | 767 |
| 19.31.12 WS.STATE | 768 |
| 19.31.13 WS.T | 769 |
| 19.31.14 WS.TDELAY1 | 770 |
| 19.31.15 WS.TDELAY2 | 771 |
| 19.31.16 WS.TDELAY3 | 772 |
| 19.31.17 WS.TDELAY4 | 773 |
| 19.31.18 WS.TIRAMP | 774 |
| 19.31.19 WS.TSTANDSTILL | 775 |
| 19.31.20 WS.VTHRESH | 776 |

19.1 About the Parameter and Command Reference Guide

This reference guide provides descriptive information about each parameter and command used in the drive firmware. Parameters and commands are used to configure the drive or to return status information from the drive using the EWV¹ terminal screen. The use of these parameters and commands to perform various drive functions is detailed in related sections of the AKD PDMM User Guide.

Drive parameter and command categories include the following:

¹Embedded Workbench Views

| | | |
|-----------------------------|-----------------------------|-----------------------------|
| AOUT Parameters (→ p. 269) | FB3 Parameters (→ p. 501) | REC Parameters (→ p. 655) |
| BODE Parameters (→ p. 281) | FBUS Parameters (→ p. 513) | REGEN Parameters (→ p. 678) |
| CS Parameters (→ p. 328) | GUI Parameters (→ p. 530) | SM Parameters (→ p. 685) |
| DIN Parameters (→ p. 333) | HWLS Parameters (→ p. 543) | STO Parameters (→ p. 698) |
| DOUT Parameters (→ p. 349) | IL Parameters (→ p. 546) | SWLS Parameters (→ p. 700) |
| DRV Parameters (→ p. 363) | LOAD Parameters (→ p. 588) | USER Parameters (→ p. 708) |
| DS402 Parameters (→ p. 446) | MOTOR Parameters (→ p. 590) | TEMP Parameters (→ p. 705) |
| FAULT Parameters (→ p. 451) | PL Parameters (→ p. 636) | VBUS Parameters (→ p. 710) |
| FB1 Parameters (→ p. 454) | | VL Parameters (→ p. 719) |
| | | WS Parameters (→ p. 756) |

A summary table of information for all parameters and commands is also available:

[Summary of Parameters and Commands](#)

For each parameter or command, this reference guide presents the following tables of information, followed by a description of the command, examples, and links to related information in the User Guide, as appropriate.

| General Information | |
|---------------------|--|
| Type | One of four types: <ul style="list-style-type: none"> • Command: Action or W/O command. • NV Parameter: R/W and stored in nonvolatile (NV) memory • R/W Parameter: Can be either read from or written to the drive. • R/O Parameter. Can only be read from the drive |
| Description | Brief description of the parameter or command and notes if the parameter or command is not active in all opmodes. |
| Units | Appropriate units |
| Range | Permissible range; multiple ranges are sometimes present. |
| Default Value | Determined at setup process time or motor ID; otherwise set to 0.010. |
| Data Type | Integer, Boolean, Float, or String |
| See Also | Links to related information such as other parameters, block diagrams, schematics, or other sections of the product manual. |
| Start Version | The minimum firmware version number required to use the parameter or command |

| Fieldbus | Index/Subindex | Object Start Version |
|---|---|---|
| Fieldbus type, such as EtherCAT COE and CANopen . | Index/subindex values for the parameter or command. The index value may be linked to the Object Dictionary for each fieldbus, if the object dictionary contains more detailed information about the object. | The minimum firmware version number required to use the fieldbus. |

Additional data types may include the following:

| Type | Description |
|-------|----------------------|
| Error | Illegal type=0 |
| b | Boolean |
| U8 | 8 x unsigned numbers |

| Type | Description |
|------|-----------------------|
| S8 | 8 x signed numbers |
| U16 | 16 x unsigned numbers |
| S16 | 16 x signed numbers |
| U32 | 32 x unsigned numbers |
| S32 | 32 x signed numbers |
| U64 | 64 x unsigned numbers |
| S64 | 64 x signed numbers |

19.1.1 Parameter and Command Naming Conventions

| Abbreviation | Term |
|--------------|----------------------------------|
| ACC | Acceleration |
| APP | Apply |
| CLR | Clear |
| CS | Controlled Stop |
| I | Current |
| D | Current d-component |
| DEC | Deceleration |
| DIR | Direction |
| DIS | Disable |
| DIST | Distance |
| EMUE | Emulated encoder |
| EN | Enable |
| ERR | Error |
| F | Fault |
| FB | Feedback |
| FF | Feedforward |
| K | Gain |
| INT | Integrator |
| LIM | Limit |
| L | Loop |
| MAX | Maximum |
| MIN | Minimum |
| N | Negative |
| NV | Nonvolatile |
| P | Position, Proportional, Positive |
| RLS | Release |
| R | Resistance |
| STATE | Status, State, Stat |

| Abbreviation | Term |
|--------------|----------------|
| THRESH | Threshold |
| T | Time |
| TMAX | Timeout |
| U | User |
| V | Velocity, Volt |
| W | Warning |

19.1.2 Summary of Parameters and Commands

This table contains an alphabetical list of parameters and commands, with a brief description for each. The parameter name and description are linked to the parameter tables. Generally speaking, all parameters and commands are active in all opmodes, with the following exceptions:

| Parameter or Command | Active in Opmodes |
|------------------------------------|------------------------------------|
| GEAR (all parameters and commands) | 2 (position) only |
| HOME (all parameters and commands) | 2 (position) only |
| MT (all parameters and commands) | 2 (position) only |
| SM.I1, SM.I2 | 0 (torque) only |
| SM.V1, SM.V2 | 1 (velocity) only |
| SM.VPM1, SM.VPM2 | 2 (position) only |
| VL (all parameters and commands) | 1 (velocity) and 2 (position) only |

| Parameter or Command | Type | Description |
|--------------------------------|------|--|
| Analog Output (AOUT) | | |
| AOUT.CUTOFF (→ p. 270) | NV | Sets the analog output low-pass filter cutoff frequency. |
| AOUT.ISCALE (→ p. 274) | NV | Sets the analog current scale factor. |
| AOUT.MODE (→ p. 275) | NV | Sets the analog output mode. |
| AOUT.OFFSET (→ p. 276) | NV | Sets the analog output offset. |
| AOUT.PSCALE (→ p. 277) | NV | Sets the analog position scale factor. |
| AOUT.VALUE (→ p. 278) | NV | Reads the analog output value. |
| AOUT.VALUEU (→ p. 279) | R/W | Sets the analog output value. |
| AOUT.VSCALE (→ p. 280) | NV | Sets the velocity scale factor for analog output. |
| Analog Output 2 (AOUT2) | | |
| AOUT2.CUTOFF (pg 1) | NV | Sets the analog output 2 low-pass filter cutoff frequency. |
| AOUT2.MODE | NV | Sets the analog output 2 mode. |
| AOUT2.OFFSET | NV | Sets the analog output 2 offset. |
| AOUT2.VALUE | NV | Reads the analog output 2 value. |
| AOUT2.VALUEU | R/W | Sets the analog output 2 value. |
| Bode plot (BODE) | | |
| BODE.EXCITEGAP (→ p. 282) | R/W | Controls how often the excitation is updated. |

| Parameter or Command | Type | Description |
|---------------------------------------|---------|--|
| BODE.FREQ (→ p. 283) | R/W | Sets the frequency of the sine excitation source. |
| BODE.IAMP (→ p. 284) | R/W | Sets current command value used during the Bode procedure. |
| BODE.IFLIMIT (→ p. 285) | R/W | Sets the current fault duration limit in seconds for the BODE.MODE 5 stability test. |
| BODE.IFTHRESH (→ p. 286) | R/W | Sets the current fault threshold for the BODE.MODE 5 stability test. |
| BODE.INJECTPOINT (→ p. 287) | R/W | Sets whether the excitation uses current or velocity excitation type. |
| BODE.MODE (→ p. 288) | R/W | Sets the mode of the excitation. |
| BODE.MODETIMER (→ p. 291) | R/W | Sets the watchdog timer of the excitation. |
| BODE.PRDEPTH (→ p. 293) | R/W | Sets the length of the PRB signal before it repeats. |
| BODE.VAMP (→ p. 294) | R/W | Sets the amplitude of the excitation when in velocity mode. |
| BODE.VFLIMIT (→ p. 295) | R/W | Sets the velocity fault duration limit (seconds) for the BODE.MODE 5 stability test |
| BODE.VFTHRESH (→ p. 296) | R/W | Sets the current fault threshold for the BODE.MODE 5 stability test. |
| Compare (CMP) | | |
| CMPx.ARM (→ p. 299) | NV | Arms/disarms the specified Compare setpoint. |
| CMPx.MODBOUND1 (→ p. 300) | NV | Sets the first modulo bound. |
| CMPx.MODBOUND2 (→ p. 302) | NV | Sets the second modulo bound. |
| CMPx.MODE (→ p. 304) | NV | Sets the mode for a Compare setpoint. |
| CMPx.MODEN (→ p. 306) | NV | CMPx.MODEN |
| CMPx.MODVALUE (→ p. 307) | R/O | Reads the modulo value. |
| CMPx.OUTMASK (→ p. 308) | NV | Sets the output mask. |
| CMPx.SETPOINT (→ p. 309) | NV | Sets the value of a Compare setpoint. |
| CMPx.SOURCE (→ p. 311) | NV | Sets the Compare source. |
| CMPx.STATE (→ p. 313) | R/O | Reads the state of a Compare setpoint. |
| CMPx.WIDTH (→ p. 314) | NV | Sets the width for a Compare setpoint. |
| CMPx.WIDTHTYPE (→ p. 316) | NV | Sets the width-type for a Compare setpoint. |
| Cogging Compensation (COGCOMP) | | |
| COGCOMP.CORRECTIONTABLE (→ p. 319) | R/W | Table of cogging compensation values. (→ p. 319) |
| COGCOMP.CORRECTIONVALUE (→ p. 320) | R/O | Cogging compensation value that is currently applied by the drive. (→ p. 320) |
| COGCOMP.EN (→ p. 321) | NV | Activates or deactivates the cogging compensation table. (→ p. 321) |
| COGCOMP.LOAD (→ p. 322) | Command | Restores the cogging compensation table from non-volatile memory. (→ p. 322) |

| Parameter or Command | Type | Description |
|---------------------------------------|---------|---|
| COGCOMP.RANGEHIGH (→ p. 323) | NV | Stores the position of the upper end of the range where cogging compensation is active on a linear axis. (→ p. 323) |
| COGCOMP.RANGELOW (→ p. 324) | NV | Stores the position of the lower end of the range where cogging compensation is active on a linear axis. (→ p. 324) |
| COGCOMP.SAVE (→ p. 325) | Command | Writes the cogging compensation table to non-volatile memory. (→ p. 325) |
| COGCOMP.TEACH (→ p. 326) | Command | Starts automatic teaching of the cogging compensation table. (→ p. 326) |
| COGCOMP.V (→ p. 327) | NV | Velocity for the move during automatic teaching of the cogging compensation table. (→ p. 327) |
| Controlled Stop (CS) | | |
| CS.DEC (→ p. 329) | NV | Sets the deceleration value for the controlled stop process. |
| CS.STATE (→ p. 330) | NV | Returns the internal status of the controlled stop process. |
| CS.TO (→ p. 331) | NV | Sets the time value for the drive velocity to be within CS.VTHRESH (→ p. 332). |
| CS.VTHRESH (→ p. 332) | NV | Sets the velocity threshold for the controlled stop. |
| Digital Input (DIN) | | |
| DIN.HCMD1 TO DIN.HCMD4 (→ p. 334) | NV | A buffer of commands to be used in digital input "command buffer" mode. |
| DIN.LCMD1 to DIN.LCMD4 (→ p. 335) | NV | A buffer of commands to be used in digital input "command buffer" mode. |
| DIN.ROTARY (→ P. 336) | R/O | Reads the rotary knob value. |
| DIN.STATES (→ P. 337) | R/O | Reads the digital input states. |
| DIN1.FILTER TO DIN7.FILTER (→ p. 338) | R/W | Filter mode for digital inputs 1 to 7 to 3. |
| DIO9.INV to DIO11.INV (→ p. 347) | R/W | Inverting the output voltage of the IO, when in the output direction. |
| DIN1.MODE TO DIN7.MODE (→ p. 340) | NV | Sets the digital input modes. |
| DIN1.PARAM TO DIN7.PARAM (→ p. 342) | R/W | Sets a value used as an extra parameter for digital inputs nodes. |
| DIN1.STATE TO DIN7.STATE (→ p. 344) | R/O | Reads a specific digital input state. |
| DIN21.FILTER to DIN31.FILTER (pg 1) | R/W | Filter mode for digital inputs 21 to 32. |
| DIN21.STATE to DIN31.STATE | R/O | Reads a specific digital input state. |
| Digital Output (DOUT) | | |
| DOUT.CTRL (→ P. 350) | NV | Sets the source of digital output(s) (firmware or fieldbus). |
| DOUT.RELAYMODE (→ p. 351) | R/W | Indicates faults relay mode. |

| Parameter or Command | Type | Description |
|--|---------|---|
| DOUT.STATES (→ P. 352) | R/O | Reads the state of the digital output(s). |
| DOUT1.MODE and DOUT2.MODE (→ p. 353) | NV | Sets the digital output mode. |
| DOUT1.PARAM AND DOUT2.PARAM (→ p. 355) | NV | Sets extra parameters for the digital output(s). |
| DOUT1.STATE AND DOUT2.STATE (→ p. 357) | R/O | Reads the digital output state. |
| DOUT1.STATEU AND DOUT2.STATEU (→ p. 358) | R/W | Sets the state of the digital output node. |
| DOUT21.STATE to DOUT31.STATE (pg 1) | R/O | Reads the digital output state. |
| DOUT21.STATEU to DOUT31.STATEU (pg 1) | R/W | Sets the state of the digital output node. |
| Drive (DRV) | | |
| DRV.ACC (→ P. 366) | NV | Describes the acceleration ramp for the velocity loop. |
| DRV.ACTIVE (→ P. 368) | R/O | Reads the enable status of an axis. |
| DRV.BLINKDISPLAY (→ P. 369) | Command | Causes the display to blink for 10 seconds. |
| DRV.BOOTTIME (→ p. 370) | R/O | Returns the time when the current session booted up. |
| DRV.CLRFAULTHIST (→ P. 372) | Command | Clears the fault history log in the NV. |
| DRV.CLRFAULTS (→ P. 373) | Command | Tries to clear all active faults in the drive. |
| DRV.CMDDELAY (→ p. 374) | R/W | Issues a delay before next command is executed. |
| DRV.CMDSOURCE (→ P. 375) | NV | Sets the command source (service, fieldbus, analog input, gearing, digital, or Bode). |
| DRV.CRASHDUMP (→ p. 376) | Command | Retrieves diagnostic information after the drive crashes. |
| DRV.DBILIMIT (→ p. 378) | NV | Sets the maximum amplitude of the current for dynamic braking. |
| DRV.DEC (→ P. 379) | NV | Sets the deceleration value for the velocity loop. |
| DRV.DIFVAR (→ p. 381) | R/O | Lists all parameters which differ from their default value. |
| DRV.DIR (→ p. 382) | R/W | Changes drive direction. |
| DRV.DIS (→ P. 384) | Command | Disables the axis (software). |
| DRV.DISMODE (→ p. 385) | NV | Selects among disable immediately or stop and then disable options. |
| DRV.DISSOURCES (→ P. 386) | R/O | Returns the possible reason for a drive disable. |
| DRV.DISSOURCESMASK (→ p. 387) | R/O | Returns a bitmap of all bits in DRV.DISSOURCES that can be set to one on a given drive. |
| DRV.DISTO (→ p. 388) | R/W | Sets the emergency timeout |
| DRV.DOWNLOADALLOWED (→ p. 389) | R/O | Returns whether the drive is in a safe state to begin a firmware download. |
| DRV.EMUEDIR (→ p. 391) | R/W | Sets the direction of the emulated encoder output (EEO) signal. |
| DRV.EMUEMODE (→ p. 392) | R/W | Sets the mode of the emulated encoder output (EEO) connector. |

| Parameter or Command | Type | Description |
|--------------------------------------|---------|---|
| DRV.EMUEMTURN (→ p. 394) | R/W | Defines the location of the index pulse on the EEO (emulated encoder output) when DRV.EMUEMODE=2. |
| DRV.EMUEPULSEWIDTH (→ p. 395) | | Sets the encoder output pulse width for modes 6 to 7. |
| DRV.EMUERES (→ p. 396) | R/W | Sets the resolution of the EEO (emulated encoder output). |
| DRV.EMUESTEPCMD (→ p. 397) | R/W | Setpoint for stepper motor output through the emulated encoder output. (→ p. 397) |
| DRV.EMUESTEPCMDPIN (→ p. 398) | R/W | Scaling factor (numerator) for the command DRV.EMUESTEPCMD. (→ p. 398) |
| DRV.EMUESTEPCMDPOUT (→ p. 399) | R/W | Scaling factor (denominator) for the command DRV.EMUESTEPCMD. (→ p. 399) |
| DRV.EMUEZOFFSET (→ p. 400) | R/W | Sets the location of the EEO (emulated encoder output) index pulse (when DRV.EMUEMODE=1). |
| DRV.EN (→ P. 401) | Command | Enables the axis (software). |
| DRV.ENDEFAULT (→ p. 402) | R/W | Sets the default state of the software enable. |
| DRV.ERRORLIST (→ p. 403) | R/O | Prints a list of all possible error codes and error text. |
| DRV.FAULTDISPLAYMODE (→ p. 404) | NV | Sets the fault display mode. |
| DRV.FAULTHIST (→ P. 405) | R/O | Reads the last 50 faults from NV memory. |
| DRV.FAULTS (→ P. 407) | R/O | Reads the active faults. |
| DRV.FAULT1 to DRV.FAULT10 (→ p. 406) | R/O | Location of fault codes for any active fault conditions. |
| DRV.HANDWHEEL | R/O | Reads the EEO input value. |
| DRV.HANDWHEELSRC (→ p. 408) | NV | Selects the feedback for handwheel operation. |
| DRV.HELP (→ P. 409) | R/O | Reads the minimum, maximum, and default values for a specific parameter or command. |
| DRV.HELPALL (→ p. 410) | R/O | Retrieves the minimum, maximum, default, and actual values for all available parameters and commands. |
| DRV.HWENABLE (→ p. 411) | R/O | Status of the hardware enable. |
| DRV.HWENDELAY (→ p. 412) | NV | Delay time between inactive Hardware Enable input and drive disable. |
| DRV.HWENMODE (→ p. 413) | R/W | Selects the action that the hardware enable digital input will perform. |
| DRV.ICONT (→ P. 414) | R/O | Reads the continuous rated current value. |
| DRV.INFO (→ P. 415) | R/O | Reads general information about the drive. |
| DRV.IPEAK (→ P. 417) | R/O | Reads the peak rated current value. |
| DRV.IZERO (→ p. 418) | R/W | Sets the current that will be used during the DRV.ZERO procedure. |
| DRV.LIST (→ P. 419) | R/O | Reads the list of available parameters and commands. |
| DRV.LOGICVOLTS (→ p. 420) | R/O | Reads the logic voltages. |

| Parameter or Command | Type | Description |
|---|---------|--|
| DRV.MOTIONDISSOURCES (→ p. 423) | R/O | Returns some causes of a motion disable. |
| DRV.NAME (→ P. 424) | NV | Sets and reads the name of the drive. |
| DRV.NVCHECK (→ p. 425) | R/O | NV Parameter Checksum |
| DRV.NVLIST (→ P. 426) | R/O | Lists the NV parameters and values from the RAM. |
| DRV.NVLOAD (→ p. 427) | W/O | Loads all data from the NV memory of the drive into the RAM parameters. |
| DRV.NVSAVE (→ P. 428) | Command | Saves the drive parameters from the RAM to the NV memory. |
| DRV.ONTIME (→ p. 429) | R/O | Returns how long the drive has been running since last power up. |
| DRV.OPMODE (→ P. 430) | NV | Sets the drive operation mode (current, velocity, or position). |
| DRV.READFORMAT (→ P. 431) | R/W | Sets the value returned to either decimal or hexadecimal. |
| DRV.REBOOT (→ p. 432) | Command | Performs a software reboot. |
| DRV.RSTVAR (→ P. 433) | Command | Sets default values in the drive without re-booting the drive and without resetting the NV memory. |
| DRV.RUNTIME (→ P. 434) | R/O | Returns how long the drive has been running since first activated. |
| DRV.SETUPREQBITS (→ p. 435) | R/O | Reads the bitwise set status of parameters that must be set before the drive can be enabled. |
| DRV.SETUPREQLIST (→ p. 437) | R/O | Reads the list of parameters that must be set before the drive can be enabled. |
| DRV.STOP (→ P. 438) | Command | This command stops all drive motion. |
| DRV.TEMPERATURES (→ p. 439) | R/O | Reads the temperature of drive components. |
| DRV.TIME | R/W | A continuous time counter in the drive. |
| DRV.TYPE (→ p. 440) | R/O | Selects the operational fieldbus on CC drive models. |
| DRV.VER (→ P. 441) | R/O | Reads the drive version. |
| DRV.VERIMAGE (→ P. 442) | R/O | Returns the version data from each image. |
| DRV.WARNINGS (→ p. 444) | R/O | Reads the active warnings. |
| DRV.WARNING1 to DRV.WARNING3 (→ p. 443) | R/O | Location of fault codes for any active warning conditions. |
| DRV.ZERO (→ p. 445) | R/W | Sets the zero mode. The procedure is activated when the drive is enabled. |
| DS402.1ADDPOSFCFEED (→ p. 447) | R/W | Sets the 1st additional feed constant - feed. (→ p. 447) |
| DS402.1ADDPOSFCSHAFTREV (→ p. 448) | R/W | Sets the 1st additional feed constant - driving shaft revolutions. (→ p. 448) |
| DS402.3ADDPOSFCFEED (→ p. 449) | R/W | "Sets the 3rd additional feed constant - feed. " (→ p. 449) |

| Parameter or Command | Type | Description |
|--|------|--|
| DS402.3ADDPOSFCSHAFTREV (→ p. 450) | R/W | Sets the 3rd additional feed constant - driving shaft revolutions. (→ p. 450) |
| Fault (FAULT) | | |
| FAULTx.ACTION (→ p. 452) | R/W | Gets/Sets the Fault Action for Fault 130, 131, 132, 134, 139, 451, and 702. |
| FAULT176.ACTION, FAULT178.ACTION (→ p. 1) | R/W | Gets/Sets the fault action for either fault 176 or 178. |
| FAULT570.ACTION (→ p. 453) | R/W | Gets/Sets the fault action for fault 570. |
| Feedback 1 (FB1) | | |
| FB1.BISSBITS (→ p. 456) | NV | Specifies the number of Biss Sensor (Position) Bits for the BiSS Mode C encoder in use. |
| FB1.CALTHRESH (→ p. 457) | NV | Sets the velocity threshold at which the auto-calibration function will continuously update error correction for sin/cos encoders. |
| FB1.DIAG (→ p. 458) | R/O | Returns feedback diagnostic information. |
| FB1.ENCRES (→ P. 459) | NV | Sets the resolution of the motor encoder. |
| FB1.FAULTS (→ p. 461) | R/O | Returns a string describing the root cause of Fault 467. |
| FB1.HALLSTATE (→ P. 463) | R/O | Reads the Hall switch values (encoder feedback only). |
| FB1.HALLSTATEU (→ p. 464) | R/O | Reads the state of Hall switch U. |
| FB1.HALLSTATEV (→ p. 465) | R/O | Reads the state of Hall switch V. |
| FB1.HALLSTATEW (→ p. 466) | R/O | Reads the state of Hall switch W. |
| FB1.IDENTIFIED (→ P. 467) | R/O | Reads the type of feedback device used by the drive/motor. |
| FB1.INITPSAVED (→ p. 468) | R/O | Reads position saved at last power down. |
| FB1.INITPSTATUS (→ p. 469) | R/O | Reads result of initial position comparison. |
| FB1.INITPWINDOW (→ p. 470) | R/W | Reads and writes size of the initial position comparison window. |
| FB1.INITSIGNED (→ p. 471) | NV | Sets initial feedback value as signed or unsigned. |
| FB1.LASTIDENTIFIED (→ p. 472) | NV | Stores the feedback type auto-detected during the last feedback initialization. Auto-detection will look for this type first. |
| FB1.MECHPOS (→ P. 473) | R/O | Reads the mechanical position. |
| FB1.MEMVER (→ P. 474) | R/O | Returns the memory feedback version. |
| FB1.MOTORPHASE (→ p. 475) | NV | Sets the motor phase of the motor associated with FB1. |
| FB1.MOTORPOLES (→ p. 476) | NV | Sets the number of motor poles associated with FB1. |
| FB1.OFFSET (→ p. 477) | NV | Sets position feedback offset. |
| FB1.ORIGIN (→ p. 478) | NV | Adds to the initial feedback position. |
| FB1.P (→ p. 480) | R/O | Reads position from the primary feedback. |
| FB1.PDIR (→ p. 481) | NV | Sets the counting direction for feedback channel 1. |
| FB1.PFIND (→ p. 482) | R/W | A procedure that allows the user to find the commutation angle for encoder feedback, which has no halls. |

| Parameter or Command | Type | Description |
|--|------|---|
| FB1.PFINDCMDU (→ p. 483) | R/W | Current value used during the phase finding procedure (PFB.PFIND=1) |
| FB1.PIN (→ p. 484) | NV | Sets gear IN for FB1.P (→ p. 480). |
| FB1.POFFSET (→ p. 485) | NV | Sets the offset for primary feedback. |
| FB1.POLES (→ P. 486) | R/O | Sets the number of feedback poles. |
| FB1.POUT (→ p. 487) | NV | Sets gear IN for FB1.P (→ p. 480). |
| FB1.PSCALE (→ p. 488) | R/W | Sets position scaling value for fieldbus transferred position objects. |
| FB1.PUNIT (→ p. 489) | NV | Sets the unit for FB1.P. |
| FB1.RESKTR (→ p. 490) | NV | Sets the resolver nominal transformation ratio. |
| FB1.RESREFPHASE (→ p. 491) | NV | Sets the electrical degrees of phase lag in the resolver. |
| FB1.SELECT (→ P. 492) | NV | Sets user entered type or identified type (-1). |
| FB1.TRACKINGCAL (→ p. 494) | NV | Controls tracking calibration algorithm. |
| FB1.USERBYTE0 to FB1.USERBYTE7 (→ p. 495) | R/W | Reads and writes data stored in eight 8 bit bytes in an Endat or Hiperface feedback device. |
| FB1.USERDWORD0 to FB1.USERDWORD1 (→ p. 497) | R/W | Reads and writes data stored in two 32 bit words in an Endat or Hiperface feedback device. |
| FB1.USERWORD0 to FB1.USERWORD3 (→ p. 499) | R/W | Reads and writes data stored in four 16 bit words in an Endat or Hiperface feedback device. |
| Feedback 3 (FB3) | | |
| FB3.DIR (→ p. 502) | NV | Sets the counting direction for the position feedback when PL.FBSOURCE = 2. (→ p. 502) |
| FB3.ENCRESES (→ p. 503) | NV | Sets the resolution of the motor encoder for feedback 3. |
| FB3.MODE (→ p. 504) | NV | Selects the type of feedback connected to X9. |
| FB3.MOTORPHASE (→ p. 505) | NV | Sets the motor phase of the motor associated with FB3. |
| FB3.MOTORPOLES (→ p. 506) | NV | Sets the number of motor poles associated with FB3. |
| FB3.P (→ p. 507) | RO | Reads position from the tertiary feedback. |
| FB3.PDIR (→ p. 508) | NV | Sets the counting direction for feedback channel 3. |
| FB3.PIN (→ p. 509) | NV | Sets gear IN for FB3.P (→ p. 507). |
| FB3.POFFSET (→ p. 510) | NV | Sets the offset for tertiary feedback. |
| FB3.POUT (→ p. 511) | NV | Sets gear IN for FB3.P (→ p. 507). |
| FB3.PUNIT (→ p. 512) | NV | Sets the unit for FB3.P. |
| Fieldbus (FBUS) | | |
| FBUS.PARAM1 TO FBUS.PARAM10 (→ p. 514) | NV | Set fieldbus specific meanings. |
| FBUS.PLLSTATE (→ p. 517) | R/O | Returns the status of the PLL |
| FBUS.PLLTHRESH (→ p. 518) | NV | Sets number of successful synchronized cycles needed to lock the PLL. |
| FBUS.PROTECTION (→ p. 519) | R/W | Controls which parameters are blocked from being accessed through telnet while a fieldbus is operational. |

| Parameter or Command | Type | Description |
|---|------|--|
| FBUS.SAMPLEPERIOD (→ p. 522) | NV | Sets fieldbus sample period. |
| FBUS.STATE (→ p. 523) | R/O | Reads the state of the fieldbus. |
| FBUS.SYNACQUIREWND (→ p. 524) | NV | Defines the time window that is used for the starting the synchronization. (→ p. 524) |
| FBUS.SYNACT (→ p. 525) | R/O | Reads actual distance from the desired sync distance. |
| FBUS.SYNCDIST (→ p. 526) | NV | Sets time target for synchronization. |
| FBUS.SYNLOCKWND (→ p. 527) | NV | Defines the time window that is used for keeping the synchronization running. (→ p. 527) |
| FBUS.SYNCWND (→ p. 528) | NV | Sets symmetrically arranged window around the desired sync distance. |
| FBUS.TYPE (→ p. 529) | R/O | Shows the active fieldbus type. |
| EWV¹ (GUI) Parameters | | |
| GUI.DISPLAY (→ P. 531) | R/O | Reads drive display data. |
| GUI.DISPLAYS (→ p. 532) | R/O | Reads drive display data. |
| GUI.PARAM01 (→ P. 533) | NV | Used by the GUI to store data. |
| GUI.PARAM02 (→ P. 534) | NV | Used by the GUI to store data. |
| GUI.PARAM03 (→ P. 535) | NV | Used by the GUI to store data. |
| GUI.PARAM04 (→ P. 536) | NV | Used by the GUI to store data. |
| GUI.PARAM05 (→ P. 537) | NV | Used by the GUI to store data. |
| GUI.PARAM06 (→ P. 538) | NV | Used by the GUI to store data. |
| GUI.PARAM07 (→ P. 539) | NV | Used by the GUI to store data. |
| GUI.PARAM08 (→ P. 540) | NV | Used by the GUI to store data. |
| GUI.PARAM09 (→ P. 541) | NV | Used by the GUI to store data. |
| GUI.PARAM10 (→ P. 542) | NV | Used by the GUI to store data. |
| Hardware Limit Switch (HWLS) | | |
| HWLS.NEGSTATE (→ p. 544) | R/O | Reads the status of the negative hardware limit switch. |
| HWLS.POSSTATE (→ p. 545) | R/O | Reads the status of the positive hardware limit switch. |
| Current Loop (IL) | | |
| IL.BUSFF (→ p. 548) | R/O | Displays the current feedforward value injected by the fieldbus. |
| IL.CMD (→ P. 549) | R/O | Reads the value of the q-component current command. |
| IL.CMDACC (→ p. 550) | R/O | Returns the commanded acceleration from the trajectory generator. |
| IL.CMDU (→ P. 551) | R/W | Sets the user current command. |
| IL.DCMD (→ p. 552) | R/O | Reads the value of the d-component current controller inside the FPGA. |
| IL.DFB (→ P. 553) | R/O | Reads the actual value of the d-component current. |

| Parameter or Command | Type | Description |
|------------------------------|------|---|
| IL.DIFOLD (→ P. 554) | R/O | Reads the drive foldback current limit. |
| IL.FB (→ P. 555) | R/O | Reads the actual value of the q-component current. |
| IL.FBSOURCE (→ p. 556) | R/W | Sets the feedback source for the current loop. Only applies when MOTOR.TYPE = 4. |
| IL.FF (→ p. 557) | R/O | Reads the current loop overall feedforward value. |
| IL.FOLDFTHRESH (→ P. 558) | NV | Reads the foldback fault level. |
| IL.FOLDFTHRESHU (→ p. 559) | NV | Sets the user value for the foldback fault level. |
| IL.FOLDWTHRESH (→ P. 560) | NV | Sets the foldback warning level. |
| IL.IFOLD (→ p. 562) | R/O | Reads the overall foldback current limit. |
| IL.IUFB (→ P. 563) | R/O | Reads the sigma-delta measured current in the u-winding of the motor. |
| IL.KACFF (→ p. 565) | R/W | Sets current loop acceleration feedforward gain value |
| IL.KBUSFF (→ p. 566) | R/W | Current loops fieldbus injected feed-forward gain |
| IL.KP (→ P. 567) | NV | Sets the proportional gain of the q-component of the PI regulator. |
| IL.KPDRATIO (→ P. 568) | NV | Sets the proportional gain of the d-component current PI-regulator as a percentage of IL.KP |
| IL.KPLOOKUPINDEX (→ p. 569) | R/W | Sets the index into the Current Loop Gain Scheduling Table. |
| IL.KPLOOKUPVALUE (→ p. 570) | R/W | Sets the value of the current loop gain scheduling index. |
| IL.KPLOOKUPVALUES (→ p. 571) | R/W | Gets the Current Loop Gain Scheduling Table. |
| IL.KPSOURCE (→ p. 572) | NV | Selects a source for the IL.KP value when MOTOR.AUTOSSET is enabled (→ p. 572) |
| IL.KVFF (→ p. 573) | R/W | Current loop velocity feed-forward gain. |
| IL.LIMITN (→ P. 574) | NV | Sets the negative user (application-specific) current limit. |
| IL.LIMITP (→ P. 575) | NV | Sets the positive user (application-specific) current limit. |
| IL.MFOLDD (→ P. 576) | NV | Sets the motor foldback maximum time at motor peak current. |
| IL.MFOLDR (→ P. 577) | R/O | Sets the motor foldback recovery time. |
| IL.MFOLDT (→ P. 578) | NV | Sets the motor foldback time constant of the exponential current drop (foldback). |
| IL.MI2T (→ p. 579) | R/O | Motor I ² t load. |
| IL.MI2TWTRESH (→ p. 580) | NV | Motor I ² t load warning threshold. |
| IL.MIFOLD (→ P. 581) | R/O | Sets the motor foldback current limit. |
| IL.MIMODE (→ p. 582) | NV | Motor protection mode. |
| IL.OFFSET (→ p. 583) | RW | A constant current command added to compensate for gravity. |
| IL.VCMD (→ P. 584) | R/O | Reads the output of the q-component PI regulator. |
| IL.VLIMIT (→ p. 585) | NV | Sets the velocity limit for opmode 0 (torque). (→ p. 585) |

| Parameter or Command | Type | Description |
|---------------------------------|---------|---|
| IL.VUFB (→ P. 586) | R/O | Reads the measured voltage on the u-winding of the motor. |
| IL.VVFB (→ P. 587) | R/O | Reads the measured voltage on the v-winding of the motor. |
| LOAD Parameters | | |
| LOAD.INERTIA (→ p. 589) | NV | Sets the load inertia. |
| Motor Parameters | | |
| MOTOR.AUTOSET (→ p. 592) | NV | Determines which drive parameters are calculated automatically. |
| MOTOR.BRAKE (→ P. 594) | NV | Sets the presence or absence of a motor brake. Can also be set to bypass F305. |
| MOTOR.BRAKEIMM (→ p. 595) | NV | Brake Immediately: in the case of a drive disable, apply the brake as soon as the power stage disables. |
| MOTOR.BRAKERLS (→ p. 596) | Command | Allows a user to release or apply the motor brake. |
| MOTOR.BRAKESTATE (→ p. 597) | R/O | Reads the actual status of the motor brake. |
| MOTOR.CTF0 (→ p. 598) | NV | Sets the thermal constant of the motor coil. |
| MOTOR.FIELDWEAKENING (→ p. 599) | R/W | Allows a surface permanent magnet motor to operate as an interior permanent magnet motor. |
| MOTOR.ICONT (→ P. 600) | NV | Sets the motor continuous current. |
| MOTOR.IDDATAVALID (→ p. 601) | R/O | Reports the status of the motor memory. |
| MOTOR.IDMAX (→ p. 602) | NV | Maximum direct axis current. |
| MOTOR.IMID (→ p. 603) | R/W | The direct-axis current set point used for induction machine closed-loop control. |
| MOTOR.IMTR (→ p. 604) | R/W | Rotor time constant. |
| MOTOR.INERTIA (→ P. 606) | NV | Sets the motor inertia. |
| MOTOR.IPEAK (→ P. 607) | NV | Sets the motor peak current. |
| MOTOR.KE (→ p. 608) | NV | Sets the motor back EMF constant. |
| MOTOR.KT (→ P. 609) | NV | Sets the torque constant of the motor. |
| MOTOR.LDLL (→ p. 610) | NV | Motor line to line direct axis inductance. |
| MOTOR.LISAT (→ p. 611) | NV | Motor quadrature axis inductance saturation characteristic. |
| MOTOR.LQLL (→ P. 612) | NV | Sets the line-to-line motor Lq. |
| MOTOR.NAME (→ P. 613) | NV | Sets the motor name. |
| MOTOR.PHASE (→ P. 614) | NV | Sets the motor phase. |
| MOTOR.PHSADV1 (→ p. 615) | NV | First order phase advance polynomial coefficient. |
| MOTOR.PHSADV2 (→ p. 616) | NV | Second order phase advance polynomial coefficient. |
| MOTOR.PITCH (→ P. 617) | NV | Sets the motor pitch. |
| MOTOR.POLES (→ P. 618) | NV | Sets the number of motor poles. |
| MOTOR.R (→ P. 619) | NV | Sets the stator winding resistance phase-phase in ohms. |
| MOTOR.RTYPE (→ p. 620) | NV | Defines the type of thermal resistor inside the motor. |

| Parameter or Command | Type | Description |
|----------------------------------|------|---|
| MOTOR.SUPPORTEDPARAMS (→ p. 621) | R/O | Lists MOTOR parameters supported by configured motor type. |
| MOTOR.TBRAKEAPP (→ P. 622) | NV | The delay time used for applying the motor brake. |
| MOTOR.TBRAKERLS (→ P. 623) | NV | The delay time used for releasing the motor brake. |
| MOTOR.TBRAKETO (→ p. 624) | NV | Brake apply timeout for vertical axis. |
| MOTOR.TEMP (→ p. 625) | R/O | Reads the motor temperature represented as the resistance of the motor PTC or NTC. |
| MOTOR.TEMPC (→ p. 626) | R/O | Reads the motor temperature in degrees Celsius. |
| MOTOR.TEMPFAULT (→ p. 627) | NV | Sets the motor temperature fault level. |
| MOTOR.TEMPWARN (→ p. 628) | NV | Sets the motor temperature warning level. |
| MOTOR.TYPE (→ P. 629) | NV | Sets the motor type. |
| MOTOR.VMAX (→ P. 631) | NV | Sets the maximum motor speed. |
| MOTOR.VOLTMAX (→ P. 632) | NV | Sets the motor maximum voltage. |
| MOTOR.VOLTMIN (→ p. 633) | NV | Sets the minimum voltage for V/f control. |
| MOTOR.VOLTRATED (→ p. 634) | NV | Sets the motor rated voltage. |
| MOTOR.VRATED (→ p. 635) | NV | Sets the motor rated velocity (not maximum velocity). |
| Position Loop (PL) | | |
| PL.CMD (→ P. 637) | NV | Reads the position command directly from the entry to the position loop. |
| PL.ERR (→ P. 638) | NV | Reads the position error present when the drive is controlling the position loop. |
| PL.ERRFTHRESH (→ p. 639) | NV | Sets the maximum position error. |
| PL.ERRMODE (→ p. 641) | R/W | Sets the type of following error warning and fault usage. |
| PL.ERRWTHRESH (→ p. 643) | NV | Sets the position error warning level. |
| PL.FB (→ P. 644) | R/O | Reads the position feedback value. |
| PL.FBSOURCE (→ p. 645) | NV | Sets the feedback source for the position loop. |
| PL.FILTERTIME (→ p. 646) | R/W | Sets the position command filter period. |
| PL.INTINMAX (→ P. 647) | NV | Limits the input of the position loop integrator by setting the input saturation. |
| PL.INTOUTMAX (→ P. 649) | NV | Limits the output of the position loop integrator by setting the output saturation. |
| PL.KI (→ P. 651) | NV | Sets the integral gain of the position loop. |
| PL.KITHRESH (→ p. 652) | R/W | Sets the velocity command threshold at which the position loop integrator turns off. (→ p. 652) |
| PL.KP (→ P. 653) | NV | Sets the proportional gain of the position regulator PID loop. |
| PL.MODP1 | R/W | Sets modulo range parameter. |
| PL.MODP2 | R/W | Sets the beginning or end modulo range parameter. |
| PL.MODPDIR | R/W | Sets the direction for absolute motion tasks. |
| PL.MODPEN | R/W | Enables the modulo position. |

| Parameter or Command | Type | Description |
|--------------------------------|---------|---|
| PL.PDELAY (→ p. 654) | R/W | Sets the delay time for PL.CMD. |
| Profinet (PN) | | |
| PN.POSSCALE (→ p. 1) | NV | Sets scaling for PNUs in Profinet which have an index greater than 2000, have the data type float, and are dependent on the position scaling parameters UNIT.PLINEAR or UNIT.PROTARY. |
| PN.TIMEOUTFTHRESH | R/O | Reads the Timeout Fault Threshold for the Profinet Communication Lost Fault (F702) |
| Recorder (REC) | | |
| REC.ACTIVE (→ P. 656) | R/O | Indicates if data recording is in progress (active). |
| REC.CH1 to REC.CH6 (→ p. 657) | R/W | Sets recording channels 1 to 6. |
| REC.DONE (→ P. 658) | R/O | Checks whether or not the recorder has finished recording. |
| REC.GAP (→ P. 659) | R/W | Specifies the gap between consecutive samples. |
| REC.NUMPOINTS (→ P. 660) | R/W | Sets the number of points to record. |
| REC.OFF (→ P. 661) | R/W | Turns the recorder OFF. |
| REC.RECPRMLIST (→ p. 662) | R/O | Reads the list of recordable parameters. |
| REC.RETRIEVE (→ P. 663) | R/O | Transfers all the recorded data to the communication channel. |
| REC.RETRIEVEDATA (→ p. 664) | R/W | Retrieves the recorded data without the header. |
| REC.RETRIEVEFRMT (→ p. 666) | R/W | Sets the format for recorded data output. |
| REC.RETRIEVEHDR (→ p. 667) | R/O | Retrieves the recorded header without the data. |
| REC.RETRIEVESIZE (→ p. 668) | R/W | Sets the number of samples that REC.RETRIEVEDATA returns. |
| REC.STOPTYPE (→ p. 669) | R/W | Sets the recorder stop type. |
| REC.TRIG (→ P. 670) | Command | Triggers the recorder. |
| REC.TRIGPARAM (→ p. 671) | R/W | Sets the parameter that triggers the recorder. |
| REC.TRIGPOS (→ p. 672) | R/W | Sets the trigger position in the recording buffer. |
| REC.TRIGPRMLIST (→ p. 674) | R/O | Reads the list of possible trigger parameters. |
| REC.TRIGSLOPE (→ P. 675) | R/W | Sets the trigger slope. |
| REC.TRIGTYPE (→ P. 676) | R/W | Sets the trigger type. |
| REC.TRIGVAL (→ P. 677) | R/W | Sets the trigger value. |
| Regen Resistor (REGEN) | | |
| REGEN.POWER (→ P. 679) | R/O | Reads regen resistor's calculated power. |
| REGEN.POWERFILTERED (→ p. 680) | R/O | Returns a filtered version of REGEN.POWER . |
| REGEN.REXT (→ P. 681) | N/V | Sets the external, user-defined regen resistor resistance. |
| REGEN.TEXT (→ p. 682) | R/W | Sets the external regen resistor thermal protection time constant. |
| REGEN.TYPE (→ P. 683) | NV | Sets the regen resistor type. |

| Parameter or Command | Type | Description |
|--|---------|---|
| REGEN.WATTEXT (→ P. 684) | R/W | Sets the regen resistor's power fault level for an external regen resistor. |
| Service Motion (SM) | | |
| SM.ACC (→ p. 686) | NV | Sets the acceleration ramp for service motion if set to greater than 0. |
| SM.DEC (→ p. 687) | NV | Sets the deceleration ramp for service motion if set to greater than 0. |
| SM.I1 (→ p. 688) | R/W | Sets service motion current 1; active in opmode 0 (torque) only. |
| SM.I2 (→ p. 689) | R/W | Sets service motion current 2; active in opmode 0 (torque) only. |
| SM.MODE (→ p. 690) | R/W | Sets the service motion mode. |
| SM.MOVE (→ p. 693) | Command | Starts the service motion. |
| SM.T1 (→ p. 694) | R/W | Sets the service motion time 1. |
| SM.T2 (→ p. 695) | R/W | Sets the service motion time 2. |
| SM.V1 (→ p. 696) | R/W | Sets service motion velocity 1; active in opmode 1 (velocity) and 2 (position). |
| SM.V2 (→ p. 697) | R/W | Sets service motion velocity 2; active in opmode 1 (velocity) and 2 (position). |
| Safe Torque Off (STO) | | |
| STO.STATE (→ p. 699) | R/O | Returns the status of the safe torque off. |
| Software Limit Switch(SWLS) | | |
| SWLS.EN (→ p. 701) | NV | Enables and disables software travel limit switches. |
| SWLS.LIMIT0 (→ p. 702) | NV | Sets the position of the software travel limit switch 0. |
| SWLS.LIMIT1 (→ p. 703) | NV | Sets the position of the software travel limit switch 0. |
| SWLS.STATE (→ p. 704) | R/O | Reads the actual status of software limit switches. |
| Temperature (TEMP) | | |
| TEMP.CONTROL (→ p. 706) | R/O | Reads the temperature of the control board temperature sensor. |
| TEMP.POWER1 to TEMP.POWER3 (→ p. 707) | R/O | Reads the temperature of the power board(s) temperature sensor(s). |
| User (USER) | | |
| USER.INT1 to USER.INT24 (→ p. 709) | R/W | Stores an integer value for future reference. |
| Bus Volatage (VBUS) | | |
| VBUS.HALFVOLT (→ p. 711) | NV | Changes voltage thresholds for -07 drives. |
| VBUS.OVFTHRESH (→ p. 712) | R/O | Reads the over voltage fault level. |
| VBUS.OVWTHRESH (→ p. 713) | N/V | Sets voltage level for over voltage warning. |
| VBUS.RMSLIMIT (→ p. 714) | R/O | Reads the limit for the bus capacitors load. |
| VBUS.UVFTHRESH (→ p. 715) | R/O | Sets the under voltage fault level. |
| VBUS.UVMODE (→ p. 716) | NV | Indicates undervoltage (UV) mode. |

| Parameter or Command | Type | Description |
|-------------------------------------|------|---|
| VBUS.UVWTHRESH (→ p. 717) | NV | Sets voltage level for undervoltage warning. |
| VBUS.VALUE (→ p. 718) | R/O | Reads DC bus voltage. |
| Velocity Loop (VL) | | |
| VL.ARPF1 TO VL.ARPF4 (→ p. 720) | R/W | Sets the natural frequency of the pole (denominator) of anti-resonance (AR) filters 1, 2, 3, and 4; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.ARPQ1 TO VL.ARPQ4 (→ p. 722) | R/W | Sets the Q of the pole (denominator) of anti-resonance (AR) filter 1; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.ARTYPE1 TO VL.ARTYPE4 (→ p. 724) | NV | Indicates the method used to calculate BiQuad coefficients; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.ARZF1 TO VL.ARZF4 (→ p. 725) | R/W | Sets the natural frequency of the zero (numerator) of anti-resonance (AR) filter 1; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.ARZQ1 TO VL.ARZQ4 (→ p. 726) | R/W | Sets the Q of the zero (numerator) of anti-resonance filter #1; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.BUSFF (→ p. 728) | R/O | Displays the velocity loop feedforward value injected by the field-bus; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.CMD (→ P. 729) | R/O | Reads the actual velocity command; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.CMDU (→ P. 730) | R/W | Sets the user velocity command; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.ERR (→ P. 731) | R/O | Sets the velocity error; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.FB (→ P. 732) | R/O | Reads the velocity feedback; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.FBFILTER (→ p. 733) | R/O | Filters VL.FB (→ p. 732) value; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.FBSOURCE (→ p. 734) | NV | Sets feedback source for the velocity loop; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.FBUNFILTERED (→ p. 735) | R/O | Reads the velocity feedback. |
| VL.FF (→ p. 736) | R/O | Displays the velocity loop overall feedforward value; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.FFDELAY (→ p. 737) | R/W | Sets the delay time for the velocity feedforward integrator component. |
| VL.GENMODE (→ P. 738) | NV | Selects mode of velocity generation (Observer, d/dt); active in opmodes 1 (velocity) and 2 (position) only. |
| VL.KBUSFF (→ p. 739) | R/W | Sets the velocity loop acceleration feedforward gain value; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.KI (→ p. 740) | NV | Sets the velocity loop integral gain for the PI controller; active in opmodes 1 (velocity) and 2 (position) only. |

| Parameter or Command | Type | Description |
|----------------------------|---------|--|
| VL.KIMODE (→ p. 741) | R/W | Sets whether the velocity Integrator is always on, or if the velocity integrator turns off any time the position loop integrator is active. (→ p. 741) |
| VL.KP (→ p. 743) | NV | Sets velocity loop proportional gain for the PI controller; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.KVFF (→ p. 745) | R/W | Sets the velocity loop velocity feedforward gain value; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.LIMITN (→ P. 746) | NV | Sets the velocity lower limit; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.LIMITP (→ P. 748) | NV | Sets the velocity high limit; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.LMJR (→ p. 749) | R/W | Sets the ratio of the estimated load moment of inertia relative to the motor moment of inertia; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.MODEL (→ p. 750) | R/O | Reads the observer velocity signal; active in opmodes 1 (velocity) and 2 (position) only. |
| VL.OBSBW (→ p. 751) | NV | Sets the bandwidth of the observer in Hz. |
| VL.OBSMODE (→ p. 752) | NV | Sets the observer operating mode. |
| VL.THRESH (→ P. 753) | NV | Sets the user over speed fault level. |
| VL.VFTHRESH (→ p. 755) | R/O | VL.VFTHRESH |
| Wake and Shake (WS) | | |
| WS.ARM (→ p. 757) | Command | Sets wake and shake to start at the next drive enable. |
| WS.CHECKMODE (→ p. 758) | R/W | Select the type of commutation check to execute after Wake and Shake finds a new commutation angle. |
| WS.CHECKT (→ p. 759) | R/W | Sets the amount of time a commutation error must be present before an error is thrown. |
| WS.CHECKV (→ p. 760) | R/W | This parameter sets the velocity threshold which has to be exceeded to activate commutation monitoring. |
| WS.DISARM (→ p. 761) | Command | Cancel ARM requests and resets wake and shake to the IDLE state. |
| WS.DISTMAX (→ p. 762) | R/W | Sets maximum movement allowed for wake and shake. |
| WS.DISTMIN (→ p. 763) | R/W | Sets the minimum movement required for wake and shake. |
| WS.FREQ (→ p. 764) | R/W | Sets the sine frequency of excitation for WS.MODE 2. |
| WS.IMAX (→ p. 765) | R/W | Sets maximum current used for wake and shake. |
| WS.MODE (→ p. 766) | R/W | Sets the method used for wake and shake. |
| WS.NUMLOOPS (→ p. 767) | R/W | Sets the number of repetitions for wake and shake. |
| WS.STATE (→ p. 768) | R/O | Reads wake and shake status. |
| WS.T (→ p. 769) | R/W | Sets wake and shake current-vector appliance time. |
| WS.TDELAY1 (→ p. 770) | NV | Delay for wake and shake timing. |
| WS.TDELAY2 (→ p. 771) | NV | Sets the delay for wake and shake timing. |
| WS.TDELAY3 (→ p. 772) | NV | Sets the delay for wake and shake between loops in mode 0. |

| Parameter or Command | Type | Description |
|---------------------------|------|--|
| WS.TDELAY4 (→ p. 773) | NV | Sets the delay time after first current-pulse injection. |
| WS.TIRAMP (→ p. 774) | R/W | Sets the ramp time for the ramp up current in Wake & Shake mode 1. |
| WS.TSTANDSTILL (→ p. 775) | R/W | Sets the calming time of the motor for Wake & Shake mode 1. |
| WS.VTHRESH (→ p. 776) | NV | Defines the maximum allowed velocity for Wake & Shake. |

19.2 AOUT Parameters

This section describes the AOUT parameters.

| | |
|--|------------|
| 19.2.1 AOUT.CUTOFF | 270 |
| 19.2.2 AOUT.DEBUGADDR | 271 |
| 19.2.3 AOUT.DEBUGDATATYPE | 272 |
| 19.2.4 AOUT.DEBUGSCALE | 273 |
| 19.2.5 AOUT.ISCALE | 274 |
| 19.2.6 AOUT.MODE | 275 |
| 19.2.7 AOUT.OFFSET | 276 |
| 19.2.8 AOUT.PSCALE | 277 |
| 19.2.9 AOUT.VALUE | 278 |
| 19.2.10 AOUT.VALUEU | 279 |
| 19.2.11 AOUT.VSCALE | 280 |

19.2.1 AOUT.CUTOFF

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the analog output low-pass filter cutoff frequency. |
| Units | Hz |
| Range | 0 to 10,000 Hz |
| Default Value | 0 Hz |
| Data Type | Float |
| Start Version | M_01-04-01 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3470h/6 | M_01-04-01-000 |

Description

AOUT.CUTOFF sets the cutoff frequency in Hz for a single-pole low-pass filter on the Analog Output.

A value of 0 Hz will turn off the filter and will allow all frequencies to pass through.

The filter can be used with all modes of Analog Output.

Related Topics

Analog Output (pg 1)

19.2.2 AOUT.DEBUGADDR

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Sets the memory address to debug. |
| Units | N/A |
| Range | 4 to 4,292,870,142 |
| Default Value | 4 |
| Data Type | Integer |
| Start Version | M_01-01-01-000 |

Description

AOUT.DEBUGADDR sets the memory address to debug when AOUT.MODE (→ p. 275) = 9 (debug mode).

Related Topics

1 Analog Output

AOUT.MODE (→ p. 275)

19.2.3 AOUT.DEBUGDATATYPE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the data type of the value to be debugged. |
| Units | N/A |
| Range | 0 to 10 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-01-01-000 |

Description

AOUT.DEBUGDATATYPE is used in AOUT.MODE (→ p. 275) = 9 (debug mode).

This parameter sets the data type of the value to be debugged according to the table below:

| Value | Data Type |
|-------|---------------------|
| 0 | Illegal type |
| 1 | Signed (1 byte) |
| 2 | Unsigned (1 byte) |
| 3 | Signed (2 bytes) |
| 4 | Unsigned (2 bytes) |
| 5 | Signed (4 bytes) |
| 6 | Unsigned (4 bytes) |
| 7 | Signed (8 bytes) |
| 8 | Unsigned (8 bytes) |
| 9 | Pointer to one byte |
| 10 | Fix shift |

Related Topics

- 1 Analog Output

19.2.4 AOUT.DEBUGSCALE

| General Information | |
|---------------------|--------------------------------------|
| Type | NV Parameter |
| Description | Sets the scale to be used for debug. |
| Units | N/A |
| Range | 0.001 to 9,223,372,036,854,775.000 |
| Default Value | 1 |
| Data Type | Float |
| Start Version | M_01-01-01-000 |

Description

AOUT.DEBUGSCALE sets the scale to be used for debug when AOUT.MODE (→ p. 275) = 9 (debug mode).

Related Topics

1 Analog Output

AOUT.MODE (→ p. 275)

19.2.5 AOUT.ISCALE

| General Information | |
|---------------------|---------------------------------------|
| Type | NV Parameter |
| Description | Sets the analog current scale factor. |
| Units | A/V |
| Range | 0.001 to 22.4 A/V |
| Default Value | 0.001 to 22.4 A/V |
| Data Type | Float |
| Start Version | M_01-01-01-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5009h/0 | M_01-00-00-000 |

Description

AOUT.ISCALE sets the analog current scale factor that scales the analog output (AOUT.VALUE) for AOUT.MODE = 4 or 5. The value entered is the motor current per 10 V of analog input or output. This value may be either higher or lower than 100%, but the actual analog I/O will be limited by the application current limit (IL.LIMITN (→ p. 574) and IL.LIMITP (→ p. 575)).

Related Topics

Analog Output (pg 1)

AOUT.ISCALE (→ p. 1)

19.2.6 AOUT.MODE

| General Information | |
|---------------------|------------------------------|
| Type | NV Parameter |
| Description | Sets the analog output mode. |
| Units | N/A |
| Range | 0 to 11 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3470h/1 | M_01-00-00-000 |

Description

AOUT.MODE sets the analog output functionality.

| AOUT.MODE | Description |
|-----------|---|
| 0 | User variable. The analog output signal is determined by the user (using AOUT.VALUEU). |
| 1 | Actual velocity. The analog signal describes the current velocity value (VL.FB). |
| 2 | Velocity error. The analog signal describes the velocity error value. |
| 3 | Velocity command. The analog signal describes the velocity command value. |
| 4 | Actual current. The analog signal describes the actual current value. |
| 5 | Current command. The analog signal describes the current command value. |
| 6 | Actual position. The analog signal describes the current position value. |
| 7 | Position error. The analog signal describes the position error value. |
| 8 | Triangle wave. The analog signal is a triangle wave (sawtooth pattern). |
| 9 | Debug mode. In this mode the user can define a drive variable to monitor via the analog output (AOUT.VALUEU). |
| 10 | Unfiltered Velocity (VL.FBUNFILTERED) |
| 11 | Filtered Velocity - 10Hz Lowpass (VL.FBFILTER) |

Example

You can use AOUT.MODE and AOUT.VALUEU to configure an output signal as follows:

```
-->AOUT.MODE 0
-->AOUT.VALUEU 5
-->AOUT.VALUEU 4.33
```

Related Topics

Analog Output (pg 1)

19.2.7 AOUT.OFFSET

| General Information | |
|---------------------|--------------------------------|
| Type | NV Parameter |
| Description | Sets the analog output offset. |
| Units | V |
| Range | -10 to +10 V |
| Default Value | 0 V |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 500Bh/0 | M_01-00-00-000 |

Description

This parameter sets the analog output offset.

Related Topics

Analog Output (pg 1)

19.2.8 AOUT.PSCALE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the analog position scale factor. |
| Units | Rotary: counts/V, rad/V, deg/V, 16-bit counts/V Linear: counts/V, mm/V, $\mu\text{m}/\text{V}$, 16-bit counts/V |
| Range | Rotary: 1 to 9,223,372,036,854,775 counts/V 0 to 13,493,026.816 rad/V 0 to 773,094,113.280 deg/V 0 to 140,737,488,355.327 16-bit counts/V Linear: 1 to 9,223,372,036,854,775 counts/V 0 to 2,147,483.648 mm/V 0 to 2,147,483,648.000 $\mu\text{m}/\text{V}$ 0 to 140,737,488,355.327 16-bit counts/V |
| Default Value | Rotary: 1 counts/V 0 rad/V 0 deg/V 0 16-bit counts/V Linear: 1 counts/V 0 rad/V 0 deg/V 0 counts 16 bit/V |
| Data Type | Float |
| Start Version | M_01-01-01-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3471h/0 | M_01-00-00-000 |

Description

AOUT.PSCALE is an analog position scale factor that scales the analog output (AOUT.VALUE (→ p. 278)) for AOUT.MODE (→ p. 275) = 6, or 7 (actual position or position error) per 10 V of analog input or output.

Related Topics

Analog Output (pg 1)

AOUT.VALUE (→ p. 278)

19.2.9 AOUT.VALUE

| General Information | |
|---------------------|--------------------------------|
| Type | R/O Parameter |
| Description | Reads the analog output value. |
| Units | V |
| Range | -10 to +10 V |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3470h/2 | M_01-00-00-000 |

Description

AOUT.VALUE reads the analog output value.

Related Topics

Analog Output (pg 1)

19.2.10 AOUT.VALUEU

| General Information | |
|---------------------|-------------------------------|
| Type | R/W Parameter |
| Description | Sets the analog output value. |
| Units | V |
| Range | -10 to +10 V |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3470h/3 | M_01-00-00-000 |

Description

AOUT.VALUEU reads/writes the analog output value when AOUT.MODE (→ p. 275) = 0 (analog output signal is determined by the user).

Related Topics

Analog Output (pg 1)

19.2.11 AOUT.VSCALE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the velocity scale factor for analog output. |
| Units | Depends on UNIT.VROTARY or UNIT.ACCLINEAR Rotary: rpm/V, rps/V, (deg/s)/V, [(custom units)/s]/V, (rad/s)/V Linear: counts/s/V, (mm/s)/V, (μm/s)/V, [(custom units)/s]/V |
| Range | Rotary: 0.060 to 60,000 rpm/V 0.001 to 1,000 rps/V 0.359 to 360,000 (deg/s)/V 0.005 to 5,000 [(custom units)/s]/V 0.006 to 6,283.186 (rad/s)/V Linear: 0.001 to 1.000 counts/s/V 0.001*MOTOR.PITCH to 1,000.000*MOTOR.PITCH (mm/s)/V 0.998*MOTOR.PITCH to 1,000,000.000*MOTOR.PITCH(μm/s)/V 0.005 to 5,000 [(custom units)/s]/V |
| Default Value | Rotary: 0.060 rpm/V 0.001 rps/V 0.359 (deg/s)/V 0.005 [(custom units)/s]/V 0.006 (rad/s)/V Linear: 0.001 counts/s/V 0.001*MOTOR.PITCH (mm/s)/V 0.998*MOTOR.PITCH (μm/s)/V 0.005 [(custom units)/s]/V |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

Variants Supported

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3470h/5 | M_01-00-00-000 |

Description

AOUT.VSCALE is an analog velocity scale factor that scales the analog output (AOUT.VALUE) for AOUT.MODE = 1, 2, or 3. The value entered is the motor velocity per 10 V of analog output. This value may be either higher or lower than the application velocity limit (VL.LIMITP or VL.LIMITN), but the actual analog I/O will be limited by VL.LIMITP or VL.LIMITN.

Related Topics

Analog Output (pg 1)

AOUT.VALUE (→ p. 278)

19.3 BODE Parameters

This section describes the BODE parameters.

| | |
|--------------------------------------|------------|
| 19.3.1 BODE.EXCITEGAP | 282 |
| 19.3.2 BODE.FREQ | 283 |
| 19.3.3 BODE.IAMP | 284 |
| 19.3.4 BODE.IFLIMIT | 285 |
| 19.3.5 BODE.IFTHRESH | 286 |
| 19.3.6 BODE.INJECTPOINT | 287 |
| 19.3.7 BODE.MODE | 288 |
| 19.3.8 BODE.MODETIMER | 291 |
| 19.3.9 BODE.PRBDEPTH | 293 |
| 19.3.10 BODE.VAMP | 294 |
| 19.3.11 BODE.VFLIMIT | 295 |
| 19.3.12 BODE.VFTHRESH | 296 |

19.3.1 BODE.EXCITEGAP

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Controls how often the excitation is updated. |
| Units | Drive samples |
| Range | 1 to 255 drive samples |
| Default Value | 2 drive samples |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5013h/0 | M_01-00-00-000 |

Description

BODE.EXCITEGAP controls how often the excitation is updated. The excitation is updated every n drive samples, where n is BODE.EXCITEGAP. For example, if BODE.EXCITEGAP = 2, then the excitation is updated every $2/(16,000 \text{ Hz}) = 1/8,000 \text{ Hz} = 0.000125 \text{ sec}$. When measuring a system, update the excitation only as often as the data is recorded.

Example

Set excitation update rate to 8,000 Hz:

```
-->BODE.EXCITEGAP 2
```

Set excitation update rate to 4,000 Hz:

```
-->BODE.EXCITEGAP 4
```

Get excitation update rate (already set to 8000 Hz):

```
-->BODE.EXCITEGAP 2
```

Related Topics

Using the Performance Servo Tuner (→ p. 104) | Using the Performance Servo Tuner: Advanced (→ p. 108)

Scope (→ p. 154)

Settings (→ p. 37)

F126 n126 (→ p. 178)

Error: Invalid Bode plot mode for this function. (→ p. 202)

19.3.2 BODE.FREQ

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the frequency of the sine excitation source. |
| Units | Hz |
| Range | 0 to 8,000 Hz |
| Default Value | 0 Hz |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

Description

BODE.FREQ sets the frequency of the sine excitation source in Hz. The sine excitation source is used to take frequency response measurements of a system.

Example

Setting up a sine excitation source of 0.2 A at 50 Hz:

```
-->BODE.INJECTPOINT 1
-->BODE.IAMP 0.2
-->BODE.FREQ 50.0
-->BODE.MODE 2
```

Related Topics

Using the Performance Servo Tuner (→ p. 104)

Using the Performance Servo Tuner: Advanced (→ p. 108)

Scope (→ p. 154)

Settings (→ p. 37)

F126 n126 (→ p. 178)

Error: Invalid Bode plot mode for this function. (→ p. 202)

BODE.MODE (→ p. 288)

BODE.INJECTPOINT (→ p. 287)

BODE.VAMP (→ p. 294)

BODE.IAMP (→ p. 284)

19.3.3 BODE.IAMP

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets current command value used during the Bode procedure. |
| Units | A |
| Range | +/- Combined drive and motor current limit |
| Default Value | 0.2 A |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5015h/0 | M_01-00-00-000 |

Description

BODE.IAMP sets the amplitude of the excitation when in current mode as set in BODE.INJECTPOINT. When using BODE.MODE (→ p. 288) = 1 and BODE.INJECTPOINT (→ p. 287) = 1, this parameter will determine the level of noise injected to commanded current value.

Example

Set the excitation current to 0.2 A:

```
-->BODE.IAMP 0.2
```

Get the excitation current (already set to 0.2 A):

```
-->BODE.IAMP 0.200 [A]
```

Related Topics

Using the Performance Servo Tuner (→ p. 104)

Using the Performance Servo Tuner: Advanced (→ p. 108)

Scope (→ p. 154)

Settings (→ p. 37)

F126 n126 (→ p. 178)

Error: Invalid Bode plot mode for this function. (→ p. 202)

BODE.INJECTPOINT (→ p. 287) | BODE.FREQ (→ p. 283)

19.3.4 BODE.IFLIMIT

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the current fault duration limit in seconds for the BODE.MODE 5 stability test. |
| Units | s |
| Range | 0.001 to 60.000 |
| Default Value | 0 |
| Data Type | Decimal |
| Start Version | M_01-02-10-000 |

Description

When BODE.MODE is set to 5, the firmware monitors the drive current (IL.CMD (→ p. 549)). When IL.CMD goes above BODE.IFTHRESH (→ p. 286), an internal counter records the length of time IL.CMD was above BODE.IFTHRESH. If the internal counter reaches BODE.IFLIMIT, Fault 133 – Instability during Autotune will be generated.

The smaller BODE.IFLIMIT, the quicker Fault 133 will be generated when IL.CMD exceeds BODE.IFLIMIT.

Example

Set BODE.IFTHRESH to 6 Amps:

```
-->BODE.IFTHRESH 6
```

Set BODE.IFLIMIT to 0.500 seconds:

```
-->BODE.IFLIMIT 0.5
```

Set BODE.MODE to 5 to enable stability detection:

```
BODE.MODE 5
```

Related Topics

Using the Performance Servo Tuner (→ p. 104)

Using the Performance Servo Tuner: Advanced (→ p. 108)

F133 (→ p. 179)

BODE.MODE (→ p. 288), BODE.MODETIMER (→ p. 291), BODE.IFTHRESH (→ p. 286), BODE.VFLIMIT (→ p. 295), BODE.VFTHRESH (→ p. 296)

19.3.5 BODE.IFTHRESH

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the current fault threshold for the BODE.MODE 5 stability test. |
| Units | A |
| Range | 0.001 to DRV.IPEAK or MOTOR.IPEAK (whichever is lowest) A |
| Default Value | 0 A |
| Data Type | Decimal |
| Start Version | M_01-02-10-000 |

Description

When BODE.MODE (→ p. 288) is set to 5, the firmware monitors the drive current (IL.CMD (→ p. 549)). When IL.CMD goes above BODE.IFTHRESH, an internal counter records the length of time IL.CMD was above BODE.IFTHRESH. If the internal counter reaches BODE.IFLIMIT (→ p. 285), Fault 133 (Instability during Autotune) is generated.

Example

Set BODE.IFTHRESH to 6 Amps:

```
-->BODE.IFTHRESH 6
```

Set BODE.IFLIMIT to 0.500 seconds:

```
-->BODE.IFLIMIT 0.5
```

Set BODE.MODE to 5 to enable stability detection:

```
BODE.MODE 5
```

Related Topics

Using the Performance Servo Tuner (→ p. 104)

Using the Performance Servo Tuner: Advanced (→ p. 108)

F133 (→ p. 179)

BODE.MODE (→ p. 288), BODE.MODETIMER (→ p. 291), BODE.VFLIMIT (→ p. 295), BODE.VFTHRESH (→ p. 296), BODE.IFLIMIT (→ p. 285)

19.3.6 BODE.INJECTPOINT

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets whether the excitation uses current or velocity excitation type. |
| Units | N/A |
| Range | 0 to 2 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5016h/0 | M_01-00-00-000 |

Description

BODE.INJECTPOINT sets whether the excitation uses current or velocity excitation type.

| BODE.INJECTPOINT | Description |
|------------------|-------------|
| 0 | None |
| 1 | Current |
| 2 | Velocity |

Example

Set BODE.INJECTPOINT to current:

```
-->BODE.INJECTPOINT 1
```

Get BODE.INJECTPOINT (already set to current):

```
-->BODE.INJECTPOINT 1
```

Related Topics

Using the Performance Servo Tuner (→ p. 104)

Using the Performance Servo Tuner: Advanced (→ p. 108)

Scope (→ p. 154)

Settings (→ p. 37)

F126 n126 (→ p. 178)

Error: Invalid Bode plot mode for this function. (→ p. 202)

BODE.IAMP (→ p. 284), BODE.MODE (→ p. 288), BODE.VAMP (→ p. 294)

19.3.7 BODE.MODE

| General Information | |
|---------------------|----------------------------------|
| Type | R/W Parameter |
| Description | Sets the mode of the excitation. |
| Units | N/A |
| Range | 0 to 4 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

Description

BODE.MODE sets the mode of the excitation. The excitation can be set to the modes shown in the table below. BODE.MODE is always set to **None** when Ethernet communication is disconnected. The peak amplitude of the excitation is set by either BODE.IAMP or BODE.VAMP (depending on BODE.INJECTPOINT).

BODE.MODE is subject to a watchdog timer (BODE.MODETIMER) as follows:

- If BODE.MODETIMER is 0, then BODE.MODE is not affected.
- If BODE.MODETIMER is set to a value greater than 0, then BODE.MODE will be set to 0 (None) after the BODE.MODETIMER time milliseconds.
- If BODE.MODE is a nonzero value, and you reset BODE.MODE to another nonzero value, you will reset the watchdog timer. This mechanism is intended to turn off the excitation signal if you lose communication with the drive.

| BODE.MODE | Description | Comments |
|-----------|-------------|---|
| 0 | None | Turns all excitation off |
| 1 | PRB | Uses Pseudo Random Binary (PRB) excitation. PRB is a signal that is always +/- peak amplitude, varying only in phase. PRB excitation results in a flat excitation frequency spectrum. PRB results in a high peak excitation amplitude, which can help minimize friction in a frequency response test. PRB excitation repeats every $(2^{BODE.PRBDEPTH}) / BODE.EXCITEGAP$ drive samples. This repetition can be used to reveal the effects of friction. |
| 2 | Sine | Uses Sine excitation |
| 3 | Noise | Uses random noise excitation. Noise is a random number generator that varies between +/- peak amplitude. |
| 4 | Offset | Sets a torque offset equal to BODE.IAMP |

Example

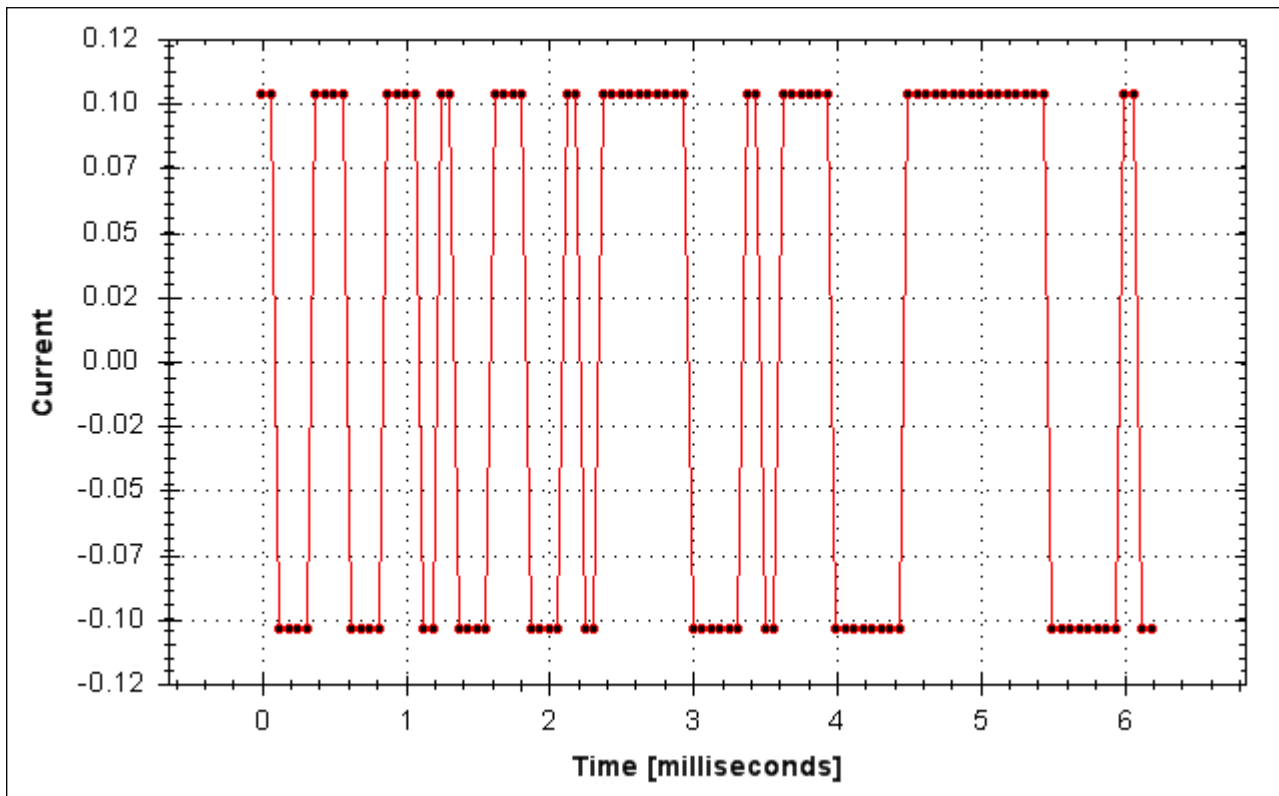
Set BODE.MODE to PRB:

```
-->BODE.MODE 1
```

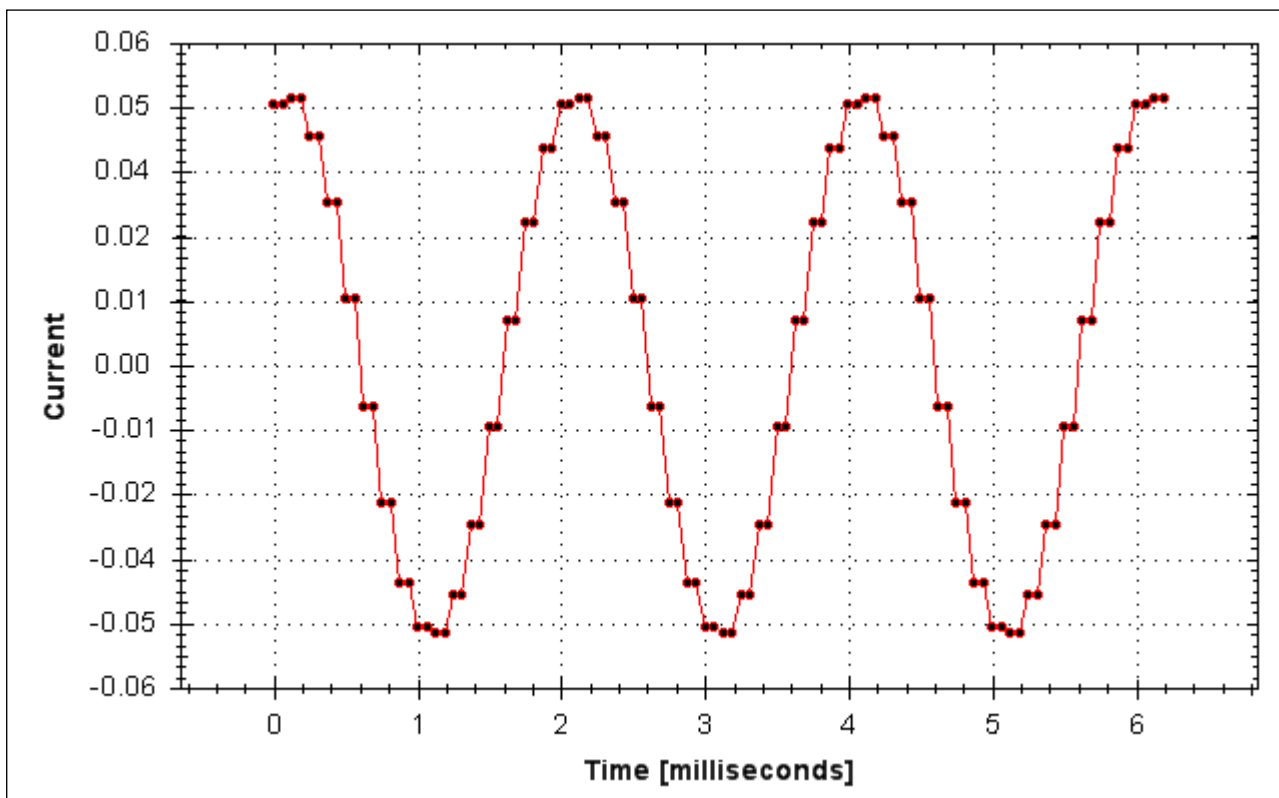
Get BODE.MODE (already set to PRB):

```
-->BODE.MODE 1
```

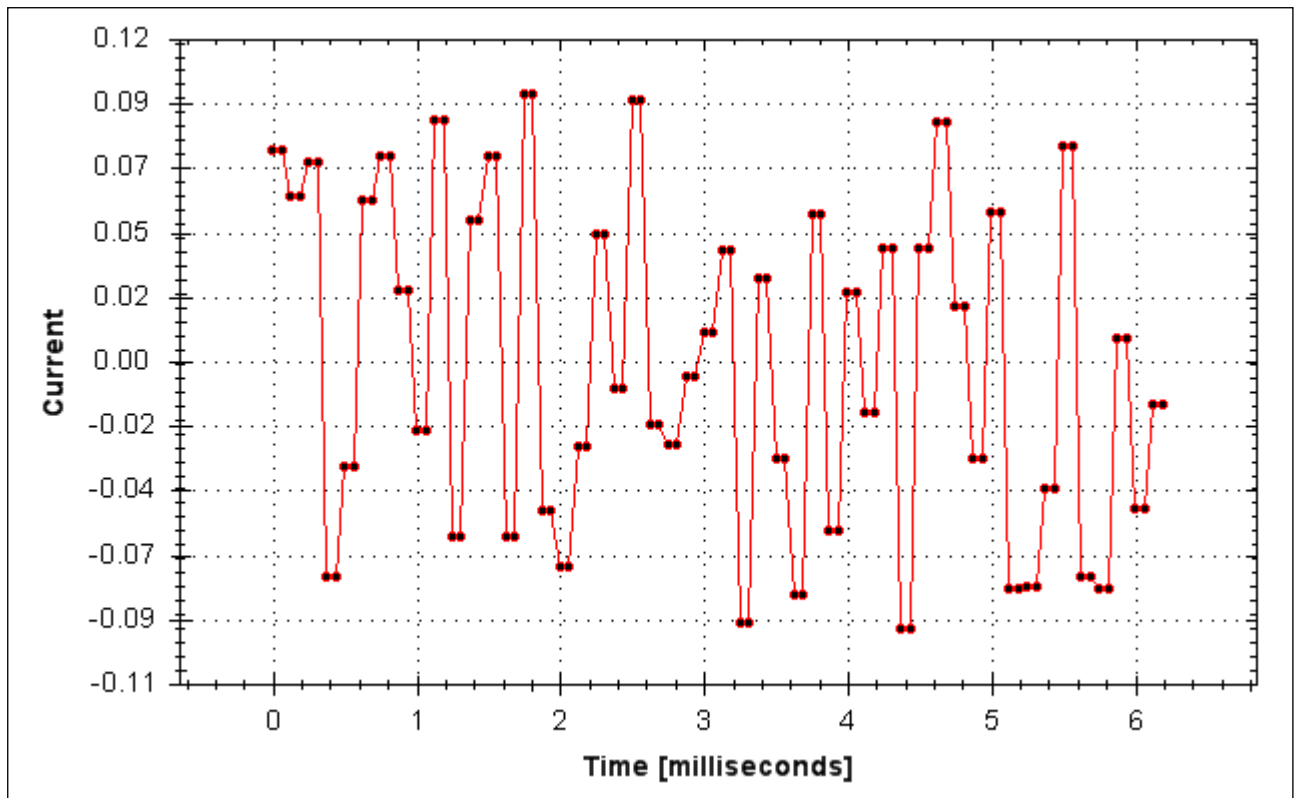
PRB excitation:



Sine excitation:



Noise excitation:



Related Topics

Using the Performance Servo Tuner (→ p. 104)

Using the Performance Servo Tuner: Advanced (→ p. 108)

Scope (→ p. 154)

Settings (→ p. 37)

F126 n126 (→ p. 178)

Error: Invalid Bode plot mode for this function. (→ p. 202)

BODE.INJECTPOINT (→ p. 287) BODE.VAMP (→ p. 294)

19.3.8 BODE.MODETIMER

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the watchdog timer of the excitation. |
| Units | ms |
| Range | 0 to 268,435,456 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_1-03-00-000 |

Variants Supported

Description

BODE.MODETIMER sets the watchdog timer for the excitation. This watchdog is used to automatically turn off the excitation of the system if communication is lost. It is highly recommended that you use the watchdog for any excitation measurements. The EWV¹ Performance Servo Tuner and Bode tool automatically use these values, requiring no action from you.

If the BODE.MODETIMER is a nonzero value, the Bode watchdog is enabled. BODE.MODE will be set to 0 (None) after the BODE.MODETIMER value elapses. To reset the watchdog timer, reset BODE.MODE to a nonzero value.

| BODE.MODETIMER | Comments |
|----------------|--|
| 0 | BODE.MODE is left at the value you set it to. |
| > 0 | Uses pseudo random binary (PRB) excitation. PRB is a signal that is always +/- peak amplitude, varying only in phase. PRB excitation results in a flat excitation frequency spectrum. PRB also results in a high peak excitation amplitude, which can help minimize friction in a frequency response test. PRB excitation repeats every $(2^{\text{BODE.PRBDEPTH}})/\text{BODE.EXCITEGAP}$ drive samples. This repetition can be used to reveal the effects of friction. |

Example

Disable BODE.MODETIMER:

```
-->BODE.MODETIMER //
Set to 0 to disable the watchdog
0
-->BODE.MODE // Observe starting state of the Bode mode
0
-->BODE.MODE 1 // Set Bode mode to PRB
-->BODE.MODE // Observe Bode mode state is the same after 0.5 seconds
1
-->BODE.MODE // Observe Bode mode state is the same after 10 seconds
```

¹Embedded Workbench Views

1

Enable BODE.MODETIMER:

-->BODE.MODETIMER 1000 // Set watchdog to 1 second

-->BODE.MODE 1 // Set Bode mode to PRB

-->BODE.MODE // Observe Bode mode state is the same after 0.5 seconds

1

-->BODE.MODE // Observe Bode mode state has been set to zero after 1.0 seconds

0

Enable and reenable BODE.MODETIMER:

-->BODE.MODETIMER 2500 // Set watchdog to 2.5 seconds

-->BODE.MODE 1 // Set Bode mode to PRB

-->BODE.MODE // Observe Bode mode state is the same after 1.5 seconds

1

-->BODE.MODE 1 // Set Bode mode to PRB, resetting the watchdog timer to the original 2.5 second value set above.

-->BODE.MODE // Observe Bode mode state is the same after 3.0 seconds after the original enabling of BODE.MODE 1

-->BODE.MODE // Observe Bode mode state has been set to zero after 4.0 seconds after the original enabling

of BODE.MODE

0

Related Topics

Using the Performance Servo Tuner (→ p. 104)

Using the Performance Servo Tuner: Advanced (→ p. 108)

Scope (→ p. 154)

Settings (→ p. 37)

F126 n126 (→ p. 178)

Error: Invalid Bode plot mode for this function. (→ p. 202)

BODE.MODE (→ p. 288)

19.3.9 BODE.PRBDDEPTH

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the length of the PRB signal before it repeats. |
| Units | NA |
| Range | 4 to 19 |
| Default Value | 19 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

Variants Supported

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5019h/0 | M_01-00-00-000 |

Description

BODE.PRBDDEPTH sets the length of the PRB signal before it repeats. This applies only when BODE.MODE = PRB. The PRB excitation will repeat after $(2^{\text{BODE.PRBDDEPTH}}) / \text{BODE.EXCITEGAP}$ drive samples.

Example

Set BODE.PRBDDEPTH to 19:

```
-->BODE.PRBDDEPTH 19
```

Get BODE.PRBDDEPTH (already set to 19):

```
-->BODE.PRBDDEPTH 19
```

Related Topics

Using the Performance Servo Tuner (→ p. 104)

Using the Performance Servo Tuner: Advanced (→ p. 108)

Scope (→ p. 154)

Settings (→ p. 37)

F126 n126 (→ p. 178)

Error: Invalid Bode plot mode for this function. (→ p. 202)

BODE.MODE (→ p. 288), BODE.INJECTPOINT (→ p. 287), BODE.IAMP (→ p. 284), BODE.VAMP (→ p. 294)

19.3.10 BODE.VAMP

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the amplitude of the excitation when in velocity mode. |
| Units | Rotary: rpm, rps, deg/s, rad/s Linear: counts/s, mm/s, $\mu\text{m/s}$ |
| Range | Rotary: 0.000 to 15,000.000 rpm 0.000 to 250.000 rps 0.000 to 90,000.000 degree/s 0.000 to 1,570.796 rad/s Linear: 0.000 to 1,073,741,824,000.000 counts/s 0.000 to 8,000.000 mm/s 0.000 to 8,000,000.000 $\mu\text{m/s}$ |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

Variants Supported

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3409h/0 | M_01-00-00-000 |

Description

BODE.VAMP sets the amplitude of the excitation when in velocity mode as set in BODE.INJECTPOINT.

Example

Set the excitation velocity to 100 RPM

```
-->BODE.VAMP 100
```

Get the excitation velocity(already set to 100 RPM)

```
-->BODE.VAMP
```

100.000 [rpm]

Related Topics

Using the Performance Servo Tuner (→ p. 104)

Using the Performance Servo Tuner: Advanced (→ p. 108)

Scope (→ p. 154)

Settings (→ p. 37)

F126 n126 (→ p. 178)

Error: Invalid Bode plot mode for this function. (→ p. 202)

BODE.MODE (→ p. 288), BODE.INJECTPOINT (→ p. 287)

19.3.11 BODE.VFLIMIT

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the velocity fault duration limit (seconds) for the BODE.MODE 5 stability test |
| Units | s |
| Range | 0.001 to 60.000 |
| Default Value | 0 |
| Data Type | Decimal |
| Start Version | M_01-02-10-000 |

Variants Supported

Description

When BODE.MODE is set to 5, the firmware monitors the drive feedback velocity VL.FB. When VL.FB goes above BODE.VFTHRESH, an internal counter records the length of time VL.FB was above BODE.VFTHRESH. If the internal counter reaches BODE.VFLIMIT, Fault 133 – Instability during Autotune will be generated.

The smaller BODE.VFLIMIT, the quicker Fault 133 will be generated when VL.FB exceeds BODE.VFLIMIT.

Example

Set BODE.VFTHRESH to 10 RPM:

```
-->BODE.VFTHRESH 10
```

Set BODE.VFLIMIT to 0.500 seconds

```
-->BODE.VFLIMIT 0.5
```

Set BODE.MODE to 5 to enable stability detection

```
-->BODE.MODE 5
```

Related Topics

Using the Performance Servo Tuner (→ p. 104)

Using the Performance Servo Tuner: Advanced (→ p. 108)

F133 (→ p. 179)

BODE.MODE, BODE.MODETIMER, BODE.IFLIMIT, BODE.IFTHRESH, BODE.VFTHRESH

19.3.12 BODE.VFTHRESH

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the current fault threshold for the BODE.MODE 5 stability test. |
| Units | Rotary: rpm, rps, deg/s, rad/s Linear: counts/s, mm/s, µm/s |
| Range | Rotary: 0.000 to 15,000.000 rpm 0.000 to 250.000 rps 0.000 to 90,000.000 deg/s 0.000 to 1,570.796 rad/s Linear: 0.000 to 1,073,741,824,000.000 counts/s 0.000 to 8,000.000 mm/s 0.000 to 8,000,000.000 µm/s 0.000 to 1,250.000 Custom Units/s |
| Default Value | 0 |
| Data Type | Decimal |
| Start Version | M_01-02-10-000 |

Variants Supported

Description

When BODE.MODE is set to 5, the firmware monitors the drive feedback velocity VL.FB. When VL.FB goes above BODE.VFTHRESH, an internal counter records the length of time VL.FB was above BODE.VFTHRESH. If the internal counter reaches BODE.VFLIMIT, Fault 133 – Instability during Autotune will be generated.

The smaller BODE.VFLIMIT, the quicker Fault 133 will be generated when VL.FB exceeds BODE.VFLIMIT.

Example

Set BODE.VFTHRESH to 10 RPM:

```
-->BODE.VFTHRESH 10
```

Set BODE.VFLIMIT to 0.500 seconds:

```
-->BODE.VFLIMIT 0.5
```

Set BODE.MODE to 5 to enable stability detection:

```
-->BODE.MODE 5
```

Related Topics

Using the Performance Servo Tuner (→ p. 104)

Using the Performance Servo Tuner: Advanced (→ p. 108)

F133 (→ p. 179)

BODE.MODE, BODE.MODETIMER, BODE.IFLIMIT, BODE.IFTHRESH, BODE.VFLIMIT

19.4 CMP Parameters

This section describes the CMP parameters.

| | |
|---------------------------------------|------------|
| 19.4.1 CMPx.ADVANCE | 298 |
| 19.4.2 CMPx.ARM | 299 |
| 19.4.3 CMPx.MODBOUND1 | 300 |
| 19.4.4 CMPx.MODBOUND2 | 302 |
| 19.4.5 CMPx.MODE | 304 |
| 19.4.6 CMPx.MODEN | 306 |
| 19.4.7 CMPx.MODVALUE | 307 |
| 19.4.8 CMPx.OUTMASK | 308 |
| 19.4.9 CMPx.SETPOINT | 309 |
| 19.4.10 CMPx.SOURCE | 311 |
| 19.4.11 CMPx.SOURCEVALUE | 312 |
| 19.4.12 CMPx.STATE | 313 |
| 19.4.13 CMPx.WIDTH | 314 |
| 19.4.14 CMPx.WIDTHTYPE | 316 |

19.4.1 CMPx.ADVANCE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the phase advance value for compare. |
| Units | ms |
| Range | 0 to 30 |
| Default Value | 0 |
| Data Type | U32 |
| Start Version | M_01-13-00-000 |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|--------------|----------------------|
| EtherCAT COE and CANopen | 53B1h/0 | CMP0.ADVANCE | M_01-13-00-000 |
| | 53B2h/0 | CMP1.ADVANCE | |

Description

Use this keyword to compensate for slow output devices such as relays or mechanical devices. The value of this parameter determines how long in advance the compare engine will trigger to compensate for a slow output.

For example, if CMP0.ADVANCE = 10, and CMPx.SETPOINT (→ p. 309) = 1,000,000 counts while the source is moving at 100,000 counts/sec (positive direction), then the compare will trigger when the source value crosses 999,000 counts. This will be 1000 counts early, or 0.01 seconds of 100,000 counts/sec.

CMPx.ADVANCE has a 0.001 ms resolution.

Related Topics

Compare Engines (→ p. 75)

19.4.2 CMPx.ARM

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Arms/disarms the specified Compare setpoint. |
| Units | N/A |
| Range | Argument 1: 0 to 7 Argument 2: 0 (Disarm) or 1 (Arm) |
| Default Value | 0 (Disarm) |
| Data Type | Argument 1: Integer Argument 2: Integer |
| Start Version | M_01-13-00-000 |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 34C0h/1 | CMP0.ARM |
| | 34C1h/1 | CMP1.ARM |
| | | M_01-13-00-000 |

Description

Use this keyword to arm or disarm a Compare setpoint. The setpoint value will only be compared against the Compare source (CMPx.SOURCE (→ p. 311)) if that setpoint is armed. If they match, an output pulse will be generated.

An error will be generated if CMPx.SOURCE, CMPx.MODBOUND1, CMPx.MODBOUND2, CMPx.WIDTH, and CMPx.MODEN are not set prior to enabling CMPx.ARM. If CMPx.SETPOINT is not set prior to enabling CMPx.ARM, the compare will trigger at the last set value, or the default value of 0 if the drive has been power cycled.

Example

The following example arms setpoint 3 for CMP1.

```
-->CMP1.ARM 3 1
```

Related Topics

Compare Engines (→ p. 75)

CMPx.SETPOINT (→ p. 309) | CMPx.SOURCE (→ p. 311)

19.4.3 CMPx.MODBOUND1

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the first modulo bound. |
| Units | Units depend on value of CMPx.SOURCE. See table below. |
| Range | -9,223,372,036,854,775,808 to 9,223,372,036,854,775,808 |
| Default Value | 9,223,372,036,854,775,808 |
| Data Type | Integer |
| Start Version | M_01-13-00-000 |

Units

| CMPx.SOURCE | Units |
|-------------|---|
| 0 (FB1) | Depends on FB1.PUNIT (→ p. 489) or custom units. |
| 1 (FB2) | Depends on FB2.PUNIT or custom units. |
| 2 (FB3) | Depends on FB2.PUNIT or custom units. |
| 3 (PL.FB) | Depends on UNIT.PLINEAR or UNIT.PROTARY Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, µm, (custom units), 16-bit counts |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 34A9h/1 | CMP0.MODBOUND1 |
| | 34B9h/1 | CMP1.MODBOUND1 |
| | | M_01-13-00-000 |

Description

Use this keyword to set the first modulo bound. CMPx.MODBOUND1 and CMPx.MODBOUND2 (→ p. 302) must both be set before the modulo functionality can be activated. There is no requirement that the second modulus bound be larger than the first modulus bound.

CMPx.MODBOUND1 must be set before CMPx.SETPOINT (→ p. 309).

Example

The following example sets the first modulo bound to one billion counts for CMP1.

```
-->CMP1.MODBOUND1 1000000000
```

Related Topics

Compare Engines (→ p. 75)

CMPx.MODBOUND2 (→ p. 302) | CMPx.MODEN (→ p. 306) | CMPx.ARM (→ p. 299)

19.4.4 CMPx.MODBOUND2

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the second modulo bound. |
| Units | Units depend on value of CMPx.SOURCE. See table below. |
| Range | -9,223,372,036,854,775,808 to 9,223,372,036,854,775,808 |
| Default Value | 9,223,372,036,854,775,808 |
| Data Type | Integer |
| Start Version | M_01-13-00-000 |

Units

| CMPx.SOURCE | Units |
|-------------|---|
| 0 (FB1) | Depends on FB1.PUNIT (→ p. 489) or custom units. |
| 1 (FB2) | Depends on FB2.PUNIT or custom units. |
| 2 (FB3) | Depends on FB2.PUNIT or custom units. |
| 3 (PL.FB) | Depends on UNIT.PLINEAR or UNIT.PROTARY Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, µm, (custom units), 16-bit counts |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 34A9h/2 | CMP0.MODBOUND2 |
| | 34B9h/2 | CMP1.MODBOUND2 |

Description

Use this keyword to set the second modulo bound. CMPx.MODBOUND1 (→ p. 300) and CMPx.MODBOUND2 must both be set before the modulo functionality can be activated. There is no requirement that the second modulus bound be larger than the first modulus bound.

CMPx.MODBOUND2 must be set before CMPx.SETPOINT (→ p. 309).

Example

The following example sets the second modulo bound to 0 counts for CMP1

```
-->CMP1.MODBOUND2 0
```

Related Topics

Compare Engines (→ p. 75)

CMPx.MODBOUND1 (→ p. 300) | CMPx.MODEN (→ p. 306) | CMPx.ARM (→ p. 299)

19.4.5 CMPx.MODE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the mode for a Compare setpoint. |
| Units | None |
| Range | Argument 1: 0 to 7 Argument 2: 0 (Single-shot) or 1 (Continuous) |
| Default Value | 1 (Continuous) |
| Data Type | Argument 1: Integer Argument 2: Integer |
| Start Version | M_01-13-00-000 |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|-------------|----------------------|
| EtherCAT COE and CANopen | 34ADh/1 | CMP0.MODE 0 | M_01-13-00-000 |
| | 34AAh/2 | CMP0.MODE 1 | |
| | 34AAh/3 | CMP0.MODE 2 | |
| | 34AAh/4 | CMP0.MODE 3 | |
| | 34AAh/5 | CMP0.MODE 4 | |
| | 34AAh/6 | CMP0.MODE 5 | |
| | 34AAh/7 | CMP0.MODE 6 | |
| | 34AAh/8 | CMP0.MODE 7 | |
| | 34BDh/1 | CMP1.MODE 0 | |
| | 34BAh/2 | CMP1.MODE 1 | |
| | 34BAh/3 | CMP1.MODE 2 | |
| | 34BAh/4 | CMP1.MODE 3 | |
| | 34BAh/5 | CMP1.MODE 4 | |
| | 34BAh/6 | CMP1.MODE 5 | |
| | 34BAh/7 | CMP1.MODE 6 | |
| | 34BAh/8 | CMP1.MODE 7 | |

Description

This keyword sets the mode for a Compare setpoint to either single-shot or continuous. Single-shot automatically disarms the setpoint after a pulse is generated. Continuous mode will keep the setpoint armed after a pulse has been generated.

Example

The following example sets the mode for setpoint 3 to single-shot.

```
-->CMP1.MODE 7 0
```

Related Topics

Compare Engines (→ p. 75)

CMPx.SETPOINT (→ p. 309)

19.4.6 CMPx.MODEN

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Enables/disables modulo functionality. |
| Units | None |
| Range | 0 to 1 |
| Default Value | 0 (Disable) |
| Data Type | Integer |
| Start Version | M_01-13-00-000 |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|------------|----------------------|
| EtherCAT COE and CANopen | 53A6h/0 | CMP0.MODEN | M_01-13-00-000 |
| | 53ADh/0 | CMP1.MODEN | |

Description

This keyword enables or disables modulo functionality on CMPx.

An error will be generated if CMPx.SOURCE, CMPx.MODBOUND1, CMPx.MODBOUND2, and CMPx.WIDTH are not set prior to enabling CMPx.MODEN. If CMPx.SETPOINT is not set prior to enabling CMPx.MODEN, the compare will trigger at the last set value, or the default value of 0 if the drive has been power cycled.

CMPx.MODEN must be set before CMPx.SETPOINT (→ p. 309).

Example

The following example enables modulo for CMP1.

```
-->CMP1.MODEN 1
```

Related Topics

Compare Engines (→ p. 75)

19.4.7 CMPx.MODVALUE

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the modulo value. |
| Units | Units depend on value of CMPx.SOURCE. See table below. |
| Range | -9,223,372,036,854,775,808 to 9,223,372,036,854,775,808 |
| Default Value | 9,223,372,036,854,775,808 |
| Data Type | Integer |
| Start Version | M_01-13-00-000 |

Units

| CMPx.SOURCE | Units |
|-------------|--|
| 0 (FB1) | Depends on FB1.PUNIT (→ p. 489) or custom units. |
| 1 (FB2) | Depends on FB2.PUNIT or custom units. |
| 2 (FB3) | Depends on FB2.PUNIT or custom units. |
| 3 (PL.FB) | Depends on UNIT.PLINEAR or UNIT.PROTARY Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, μ m, (custom units), 16-bit counts |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 34A8h/0 | CMP0.MODVALUE |
| | 34B8h/0 | CMP1.MODVALUE |
| | | M_01-13-00-000 |

Description

Use this keyword to read the modulo value.

Related Topics

Compare Engines (→ p. 75)

CMPx.MODBOUND2 (→ p. 302) | CMPx.MODBOUND2 (→ p. 302) | CMPx.MODEN (→ p. 306) | CMPx.ARM (→ p. 299)

19.4.8 CMPx.OUTMASK

| General Information | |
|---------------------|-----------------------|
| Type | NV Parameter |
| Description | Sets the output mask. |
| Units | None |
| Range | 0 to 255 |
| Default Value | 0 (Inactive) |
| Data Type | Integer |
| Start Version | M_01-13-00-000 |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|--------------|----------------------|
| EtherCAT COE and CANopen | 5394h/0 | CMP0.OUTMASK | M_01-13-00-000 |
| | 539Bh/0 | CMP1.OUTMASK | |

Description

This keyword sets the binary output disable mask between the setpoint and the drive's output pin. Note the internal setpoint state remains valid, only the external output is blocked.

Example

The following example sets an outmask to 7 (binary 111) to block setpoint states 0, 1, and 2 from asserting the drive output pin.

```
-->CMP0.OUTMASK 7
```

Related Topics

Compare Engines (→ p. 75)

CMPx.SETPOINT (→ p. 309)

19.4.9 CMPx.SETPOINT

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the value of a Compare setpoint. |
| Units | Units depend on value of CMPx.SOURCE. See table below. |
| Range | Argument 1: 0 to 7 Argument 2: -9,223,372,036,854,775,808 to 9,223,372,036,854,775,808 |
| Default Value | 9,223,372,036,854,775,808 |
| Data Type | Argument 1: Integer Argument 2: Integer |
| Start Version | M_01-13-00-000 |

Units

| CMPx.SOURCE | Units |
|-------------|---|
| 0 (FB1) | Depends on FB1.PUNIT (→ p. 489) or custom units. |
| 1 (FB2) | Depends on FB2.PUNIT or custom units. |
| 2 (FB3) | Depends on FB2.PUNIT or custom units. |
| 3 (PL.FB) | Depends on UNIT.PLINEAR or UNIT.PROTARY Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, µm, (custom units), 16-bit counts |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | Object Start Version | |
|--------------------------|----------------|----------------------|----------------|
| EtherCAT COE and CANopen | 34AAh/1 | CMP0.SETPOINT 0 | M_01-13-00-000 |
| | 34AAh/2 | CMP0.SETPOINT 1 | |
| | 34AAh/3 | CMP0.SETPOINT 2 | |
| | 34AAh/4 | CMP0.SETPOINT 3 | |
| | 34AAh/5 | CMP0.SETPOINT 4 | |
| | 34AAh/6 | CMP0.SETPOINT 5 | |
| | 34AAh/7 | CMP0.SETPOINT 6 | |
| | 34AAh/8 | CMP0.SETPOINT 7 | |
| | 34BAh/1 | CMP1.SETPOINT 0 | |
| | 34BAh/2 | CMP1.SETPOINT 1 | |
| | 34BAh/3 | CMP1.SETPOINT 2 | |
| | 34BAh/4 | CMP1.SETPOINT 3 | |
| | 34BAh/5 | CMP1.SETPOINT 4 | |
| | 34BAh/6 | CMP1.SETPOINT 5 | |
| | 34BAh/7 | CMP1.SETPOINT 6 | |
| | 34BAh/8 | CMP1.SETPOINT 7 | |

Description

Use this keyword to set the position value for a Compare setpoint. This position value will be compared against a Compare source (CMPx.SOURCE (→ p. 311)). If they match, an output pulse is generated. A setpoint also requires a width (CMPx.WIDTH (→ p. 314)) and width type (CMPx.WIDTHTYPE (→ p. 316)).

A setpoint must be armed (CMPx.ARM (→ p. 299)) before it is active. If a setpoints value, width, or width type are changed after it has been armed, that setpoint will be automatically disarmed.

Example

The following example sets the value of setpoint 1 to one billion for CMP1.

```
-->CMP1.SETPOINT 1 1000000000
```

Related Topics

Compare Engines (→ p. 75)

19.4.10 CMPx.SOURCE

| General Information | |
|---------------------|--------------------------|
| Type | NV Parameter |
| Description | Sets the Compare source. |
| Units | None |
| Range | 0 to 3 |
| Default Value | 3 |
| Data Type | Integer |
| Start Version | M_01-13-00-000 |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|-------------|----------------------|
| EtherCAT COE and CANopen | 538Fh/0 | CMP0.SOURCE | M_01-13-00-000 |
| | 5390h/0 | CMP1.SOURCE | |

Description

Setpoints (CMPx.SETPOINT (→ p. 309)) are compared against Compare source values. The setpoint must be armed (CMPx.ARM (→ p. 299)) for it to be compared against a source. If they match, an output pulse is generated.

| Value | Compare Source |
|-------|---|
| 0 | Feedback 1 (FB1) |
| 1 | Feedback 2 (FB2) |
| 2 | Feedback 3 (FB3) |
| 3 | Position Loop Feedback (PL.FB (→ p. 644)) |

The source must be set before CMPx.SETPOINT (→ p. 309).

Example

The following example sets the source to PL.FB for CMP0.

```
-->CMP0.SOURCE 3
```

Related Topics

Compare Engines (→ p. 75)

19.4.11 CMPx.SOURCEVALUE

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the current value of the selected source for CMPx.SOURCE. |
| Units | Units depend on value of CMPx.SOURCE. See table below. |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| Start Version | M_01-13-00-000 |

Units

| CMPx.SOURCE | Units |
|-------------|---|
| 0 (FB1) | Depends on FB1.PUNIT (→ p. 489) or custom units. |
| 1 (FB2) | Depends on FB2.PUNIT or custom units. |
| 2 (FB3) | Depends on FB2.PUNIT or custom units. |
| 3 (PL.FB) | Depends on UNIT.PLINEAR or UNIT.PROTARY Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, µm, (custom units), 16-bit counts |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

Description

Reads the current value of the selected source for CMPx.SOURCE.

Example

If CMP0.SOURCE = 3 (PL.FB), CMP0.SOURCEVALUE will return the current value of PL.FB.

Related Topics

Compare Engines (→ p. 75) | CMPx.SOURCEVALUE

19.4.12 CMPx.STATE

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the state of a Compare setpoint. |
| Units | None |
| Range | 0 to 7 |
| Default Value | 0 (Inactive) |
| Data Type | Integer |
| Start Version | M_01-13-00-000 |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|------------|----------------------|
| EtherCAT COE and CANopen | 34C0h/2 | CMP0.STATE | M_01-13-00-000 |
| | 34C1h/2 | CMP1.STATE | |

Description

Use this keyword to read the state of a Compare setpoint. The state indicates whether a compare pulse is being generated. If the state is 1, then a pulse is being generated. If the state is 0, no pulse is being generated.

Related Topics

Compare Engines (→ p. 75)

CMPx.SETPOINT (→ p. 309)

19.4.13 CMPx.WIDTH

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the width for a Compare setpoint. |
| Units | Position counts or microseconds depending on CMPx.WIDTHTYPE (→ p. 316). If CMPx.WIDTHTYPE = 0 (position counts), then units depend on CMPx.SOURCE (see table below). |
| Range | Argument 1: 0 to 7 Argument 2: 0 to 18,446,744,073,709,551,615 (position counts) or 0 to 65,535,000 (microseconds) |
| Default Value | 0 |
| Data Type | Argument 1: Integer Argument 2: Integer |
| Start Version | M_01-13-00-000 |

Units

If CMPx.WIDTHTYPE = 0 (position counts), determine units with the following table.

| CMPx.SOURCE | Units |
|-------------|---|
| 0 (FB1) | Depends on FB1.PUNIT (→ p. 489) or custom units. |
| 1 (FB2) | Depends on FB2.PUNIT or custom units. |
| 2 (FB3) | Depends on FB2.PUNIT or custom units. |
| 3 (PL.FB) | Depends on UNIT.PLINEAR or UNIT.PROTARY Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, µm, (custom units), 16-bit counts |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|--------------|----------------------|
| EtherCAT COE and CANopen | 34ABh/1 | CMP0.WIDTH 0 | M_01-13-00-000 |
| | 34AAh/2 | CMP0.WIDTH 1 | |
| | 34AAh/3 | CMP0.WIDTH 2 | |
| | 34AAh/4 | CMP0.WIDTH 3 | |
| | 34AAh/5 | CMP0.WIDTH 4 | |
| | 34AAh/6 | CMP0.WIDTH 5 | |
| | 34AAh/7 | CMP0.WIDTH 6 | |
| | 34AAh/8 | CMP0.WIDTH 7 | |
| | 34BBh/1 | CMP1.WIDTH 0 | |
| | 34BAh/2 | CMP1.WIDTH 1 | |
| | 34BAh/3 | CMP1.WIDTH 2 | |
| | 34BAh/4 | CMP1.WIDTH 3 | |
| | 34BAh/5 | CMP1.WIDTH 4 | |
| | 34BAh/6 | CMP1.WIDTH 5 | |
| | 34BAh/7 | CMP1.WIDTH 6 | |
| | 34BAh/8 | CMP1.WIDTH 7 | |

Description

Use this keyword to set the width for a Compare setpoint. The width defines for how much time or for how many position counts a pulse will be generated once a setpoint is active.

If a pulse width's type is in microseconds, the resolution is as follows:

| Width | Resolution |
|----------------------------|------------|
| 0 to 65,535 μ s | μ s |
| 65,536 μ s to 65.535 s | ms |

CMPx.WIDTH must be set before CMPx.SETPOINT (→ p. 309).

Example

The following example sets the width type of setpoint 7 to microseconds, and the width of setpoint 7 to 1000 microseconds.

```
-->CMP0.WIDTHTYPE 7 1
```

```
-->CMP0.WIDTH 7 1000
```

Related Topics

Compare Engines (→ p. 75)

CMPx.WIDTHTYPE (→ p. 316)

19.4.14 CMPx.WIDTHTYPE

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the width-type for a Compare setpoint. |
| Units | None |
| Range | Argument 1: 0 to 7 Argument 2: 0 (position) to 1 (time) |
| Default Value | 0 |
| Data Type | Argument 1: Integer Argument 2: Integer |
| Start Version | M_01-13-00-000 |

Hardware Supported

Compare functionality is supported on hardware revisions D, DB, DA, and higher depending on the drive type. Check the label on the side of the drive to ensure the AKD PDMM meets one of the following requirements.

| AKD PDMM Type | Hardware Revision Supported |
|---------------|-----------------------------|
| AKD PDMM | Revision D or higher |
| AKD PDMM | Revision DB or higher |
| AKD PDMM-T-IC | Revision DA or higher |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|------------------|----------------------|
| EtherCAT COE and CANopen | 34ACh/1 | CMP0.WIDTHTYPE 0 | M_01-13-00-000 |
| | 34AAh/2 | CMP0.WIDTHTYPE 1 | |
| | 34AAh/3 | CMP0.WIDTHTYPE 2 | |
| | 34AAh/4 | CMP0.WIDTHTYPE 3 | |
| | 34AAh/5 | CMP0.WIDTHTYPE 4 | |
| | 34AAh/6 | CMP0.WIDTHTYPE 5 | |
| | 34AAh/7 | CMP0.WIDTHTYPE 6 | |
| | 34AAh/8 | CMP0.WIDTHTYPE 7 | |
| | 34BCh/1 | CMP1.WIDTHTYPE 0 | |
| | 34BAh/2 | CMP1.WIDTHTYPE 1 | |
| | 34BAh/3 | CMP1.WIDTHTYPE 2 | |
| | 34BAh/4 | CMP1.WIDTHTYPE 3 | |
| | 34BAh/5 | CMP1.WIDTHTYPE 4 | |
| | 34BAh/6 | CMP1.WIDTHTYPE 5 | |
| | 34BAh/7 | CMP1.WIDTHTYPE 6 | |
| | 34BAh/8 | CMP1.WIDTHTYPE 7 | |

Description

Use this keyword to set the width-type for a Compare setpoint. The width-type determines whether the width of the Compare pulse is in position counts or units of time (microseconds).

Example

The following example sets the width type of setpoint 7 to microseconds, and the width of setpoint 7 to 1000 microseconds.

```
-->CMP0.WIDTHTYPE 7 1
```

```
-->CMP0.WIDTH 7 1000
```

Related Topics

Compare Engines (→ p. 75)

CMPx.WIDTH (→ p. 314)

19.5 COGCOMP Parameters

This section describes the COGCOMP parameters.

| | |
|---|------------|
| 19.5.1 COGCOMP.CORRECTIONTABLE | 319 |
| 19.5.2 COGCOMP.CORRECTIONVALUE | 320 |
| 19.5.3 COGCOMP.EN | 321 |
| 19.5.4 COGCOMP.LOAD | 322 |
| 19.5.5 COGCOMP.RANGEHIGH | 323 |
| 19.5.6 COGCOMP.RANGELOW | 324 |
| 19.5.7 COGCOMP.SAVE | 325 |
| 19.5.8 COGCOMP.TEACH | 326 |
| 19.5.9 COGCOMP.V | 327 |

19.5.1 COGCOMP.CORRECTIONTABLE

This content is not applicable to KAS, using either a PAC or a PDMM.

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Table of cogging compensation values. |
| Units | mA |
| Range | Argument 1: 0 to 8,191 Argument 2: $1.628 * \text{DRV.IPEAK}$ (→ p. 417) |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-14-06-001 |

Description

Reads or writes to the cogging compensation table.

Examples

Reads the full table:

```
COGCOMP.CORRECTIONTABLE
```

Reads the compensation value for index "x":

```
COGCOMP.CORRECTIONTABLE x
```

Writes the value "y" to the table at index "x":

```
COGCOMP.CORRECTIONTABLE x y
```

Related Topics

19.5.2 COGCOMP.CORRECTIONVALUE

This content is not applicable to KAS, using either a PAC or a PDMM.

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Cogging compensation value that is currently applied by the drive. |
| Units | mA |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-14-06-001 |

Description

If COGCOMP.EN (→ p. 321) is set to 1, this parameter returns the actual current feedforward compensation value based on the current feedback position, which is derived from the cogging compensation table (COGCOMP.CORRECTIONTABLE (→ p. 319)).

If COGCOMP.EN is set to 0, this parameter will always return 0.

Related Topics

19.5.3 COGCOMP.EN

This content is not applicable to KAS, using either a PAC or a PDMM.

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Activates or deactivates the cogging compensation table. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-14-06-001 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 540Fh/0 | M_01-14-06-001 |

Description

If the COGCOMP.EN is set to one, the current regulator will add a feedforward current depending on the actual feedback position (PL.FB (→ p. 644)) stored in the cogging compensation correction value (COGCOMP.CORRECTIONTABLE (→ p. 319)).

Related Topics

19.5.4 COGCOMP.LOAD

This content is not applicable to KAS, using either a PAC or a PDMM.

| General Information | |
|---------------------|---|
| Type | Command |
| Description | Restores the cogging compensation table from non-volatile memory. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-14-07-000 |

Description

This parameter will overwrite the actual entries in cogging compensation table (COGCOMP.CORRECTIONTABLE (→ p. 319)) with the stored values in NV memory.

Related Topics

19.5.5 COGCOMP.RANGEHIGH

This content is not applicable to KAS, using either a PAC or a PDMM.

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Stores the position of the upper end of the range where cogging compensation is active on a linear axis. |
| Units | Position units |
| Range | -9,223,372,036,854,774 [counts] to 9,223,372,036,854,774 [counts] or equivalent in position units |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-14-06-001 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5416h/0 | M_01-14-06-001 |

Description

This parameter stores the position of the upper end of the range where cogging compensation is active on a linear axis.

When the value is changed while compensation values are configured, the compensation values will be redistributed over the new range, which will most likely lead to incorrect compensation.

Before starting COGCOMP.TEACH (→ p. 326) on a linear axis, the value in COGCOMP.RANGEHIGH has to be larger than COGCOMP.RANGELOW (→ p. 324).

Related Topics

19.5.6 COGCOMP.RANGELOW

This content is not applicable to KAS, using either a PAC or a PDMM.

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Stores the position of the lower end of the range where cogging compensation is active on a linear axis. |
| Units | Position units |
| Range | -9,223,372,036,854,774 [counts] to 9,223,372,036,854,774 [counts] or equivalent in position units |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-14-06-001 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5418h/0 | M_01-14-06-001 |

Description

This parameter stores the position of the lower end of the range where cogging compensation is active on a linear axis.

When the value is changed while compensation values are configured, the compensation values will be redistributed over the new range, which will most likely lead to incorrect compensation.

Before starting COGCOMP.TEACH (→ p. 326) on a linear axis, the value in COGCOMP.RANGELOW has to be smaller than COGCOMP.RANGEHIGH (→ p. 323).

Related Topics

19.5.7 COGCOMP.SAVE

This content is not applicable to KAS, using either a PAC or a PDMM.

| General Information | |
|---------------------|---|
| Type | Command |
| Description | Writes the cogging compensation table to non-volatile memory. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-14-07-000 |

Description

This parameter saves the current cogging compensation table (COGCOMP.CORRECTIONTABLE (→ p. 319)) from RAM to NV memory. On next drive boot, the cogging compensation table will be initialized with the stored entries from NV.

Related Topics

19.5.8 COGCOMP.TEACH

This content is not applicable to KAS, using either a PAC or a PDMM.

| General Information | |
|---------------------|--|
| Type | Command |
| Description | Starts automatic teaching of the cogging compensation table. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-14-07-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 540Fh/0 | M_01-14-07-000 |

Description

To execute this command the drives operating mode needs to be velocity, and the command source needs to be service.

For rotary applications:

The axis will move autonomously around one turn with the defined velocity in COGCOMP.V (→ p. 327).

For linear applications:

The axis will move autonomously in the range defined in COGCOMP.RANGELOW (→ p. 324) and COGCOMP.RANGEHIGH (→ p. 323). The start position of the axis must be below COGCOMP.RANGELOW. The move will be done in velocity mode and the drive will ramp down the velocity when COGCOMP.RANGEHIGH is reached. This means the final end position is defined by the COGCOMP.RANGEHIGH position and deceleration ramp.

For both applications:

During this move, the torque distortion due to cogging of the motor will be measured. While the move is executed, the DRV.MOTIONSTAT bit 22 is set. After the move has been finished and the correction table has been calculated, bit 22 is cleared.

NOTE

To achieve a good result, it's recommended to use very stiff tuning for this move and have no load attached to the axis. Otherwise, the move might create implausible entries for the cogging correction table.

Related Topics

19.5.9 COGCOMP.V

This content is not applicable to KAS, using either a PAC or a PDMM.

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Velocity for the move during automatic teaching of the cogging compensation table. |
| Units | Depends on UNIT.VROTARY or UNIT.VLINEAR Rotary: rpm, rps, deg/s, (custom units)/s, rad/s Linear: counts/s, mm/s, $\mu\text{m/s}$, (custom units)/s |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| Start Version | M_01-14-06-001 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5414h/0 | M_01-14-06-001 |

Description

Velocity for the move during automatic teaching of the cogging compensation table.

Related Topics

COGCOMP.TEACH (→ p. 326)

19.6 CS Parameters

Controlled stop (CS) parameters set the values for the controlled stop process.

| | |
|--------------------------------|------------|
| 19.6.1 CS.DEC | 329 |
| 19.6.2 CS.STATE | 330 |
| 19.6.3 CS.TO | 331 |
| 19.6.4 CS.VTHRESH | 332 |

19.6.1 CS.DEC

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the deceleration value for the controlled stop process. |
| Units | Rotary: rps/s, rpm/s, deg/s ² , rad/s ² Linear: counts/s ² , mm/s ² , μm/s ² |
| Range | Rotary: 0.002 to 833,333.333 rps/s 0.112 to 50,000,000.000 rpm/s 0.009 to 300,000,000.000 deg/s ² 0.012 to 5,235,987.968 rad/s ² Linear: 16,000.000 to 3,579,139,408,000.000 counts/s ² 0.031*MOTOR.PITCH (→ p. 617) to 833333.333*MOTOR.PITCH (→ p. 617) mm/s ² 30.994*MOTOR.PITCH (→ p. 617) to 833333333.333*MOTOR.PITCH (→ p. 617) μm/s ² |
| Default Value | Rotary: 166.669 rps/s 10,000.000 rpm/s 60,000.000 deg/s ² 1,047.2 rad/s ² Linear: 715,840,000.000 counts/s ² 166.714*MOTOR.PITCH (→ p. 617)MOTOR.PITCH (→ p. 617) mm/s ² 166,714.191*MOTOR.PITCH (→ p. 617)MOTOR.PITCH (→ p. 617) μm/s ² |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3440h/1 | M_01-00-00-000 |

Description

This parameter sets the deceleration value for the controlled stop process.

Related Topics

Controlled Stop (→ p. 80)

Digital Inputs and Outputs (pg 1)

Fault and Warning Messages (→ p. 177)

CS.VTHRESH (→ p. 332), CS.TO (→ p. 331), DRV.DIS, DIN1.MODE TO DIN7.MODE (→ p. 340), DRV.DISM0DE (→ p. 385), DRV.DISSOURCES (→ p. 386)

19.6.2 CS.STATE

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Returns the internal status of the controlled stop process. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3441h/0 | M_01-00-00-000 |

Description

CS.STATE returns the internal state machine value of the controlled stop.

0 = controlled stop is not occurring.

1 = controlled stop is occurring

Related Topics

Controlled Stop (→ p. 80)

Digital Inputs and Outputs (pg 1)

Fault and Warning Messages (→ p. 177)

CS.DEC (→ p. 329), CS.VTHRESH (→ p. 332), CS.TO (→ p. 331) DRV.DISMODE (→ p. 385),
 DRV.DISSOURCES (→ p. 386)

19.6.3 CS.TO

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the time value for the drive velocity to be within CS.VTHRESH (→ p. 332). |
| Units | ms |
| Range | 1 to 30,000 ms |
| Default Value | 6 ms |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3440h/3 | M_01-00-00-000 |

Description

CS.TO is the time value for the drive velocity to be within CS.VTHRESH (→ p. 332) before the drive disables.

Example

Set time value to 100 ms:

```
-->CS.TO 100
```

Related Topics

Controlled Stop (→ p. 80)

Digital Inputs and Outputs

Fault and Warning Messages (→ p. 177)

CS.DEC (→ p. 329), CS.VTHRESH (→ p. 332), CS.STATE, DRV.DIS, DIN1.MODE TO DIN7.MODE (→ p. 340), DRV.DISMODE (→ p. 385), DRV.DISSOURCES (→ p. 386)

19.6.4 CS.VTHRESH

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the velocity threshold for the controlled stop. |
| Units | rpm, rps, deg/s |
| Range | Rotary: 0.000 to 15,000.000 rpm 0.000 to 250.000 rps 0.000 to 90,000.000 deg/s 0.000 to 1,570.796 rad/s Linear: 0.000 to 1,073,741,824,000.000 counts/s 0.000 to 8,000.000 mm/s 0.000 to 8,000,000.000 µm/s |
| Default Value | 5 rpm |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3440h/2 | M_01-00-00-000 |

Description

CS.VTHRESH is the velocity threshold for the controlled stop algorithm.

Example

Set velocity threshold for controlled stop at 100 rpm:

```
-->CS.VTHRESH 100
```

Related Topics

Controlled Stop (→ p. 80)

Digital Inputs and Outputs (pg 1)

Fault and Warning Messages (→ p. 177)

CS.DEC (→ p. 329), CS.TO (→ p. 331), CS.STATE (→ p. 330), DRV.DIS (→ p. 384), DIN1.MODE TO DIN7.MODE (→ p. 340), DRV.DISM MODE (→ p. 385), DRV.DISSOURCES (→ p. 386)

19.7 DIN Parameters

This section describes the DIN parameters.

| | |
|--|------------|
| 19.7.1 DIN.HCMD1 TO DIN.HCMD4 | 334 |
| 19.7.2 DIN.LCMD1 to DIN.LCMD4 | 335 |
| 19.7.3 DIN.ROTARY | 336 |
| 19.7.4 DIN.STATES | 337 |
| 19.7.5 DIN1.FILTER TO DIN7.FILTER | 338 |
| 19.7.6 DIN1.INV to DIN7.INV | 339 |
| 19.7.7 DIN1.MODE TO DIN7.MODE | 340 |
| 19.7.8 DIN1.PARAM TO DIN7.PARAM | 342 |
| 19.7.9 DIN1.STATE TO DIN7.STATE | 344 |
| 19.7.10 DIN9.STATE to DIN11.STATE | 345 |

19.7.1 DIN.HCMD1 TO DIN.HCMD4

| General Information | |
|---------------------|---|
| Type | NV parameter |
| Description | A buffer of commands to be used in digital input "command buffer" mode. |
| Units | None |
| Range | A string of up to 128 characters |
| Default Value | <Empty> |
| Data Type | String |
| Start Version | M_01-02-08-000 |

Description

DIN.HCMDx sets the string of commands to be used in the digital input mode command buffer. Digital input mode **9-Command buffer** can execute four different sets of command buffers.

Each set of command buffers contains two buffers:

- High buffer: Executes upon a rising edge of a digital input.
- Low buffer: Executes upon a falling edge of a digital input.

DIN.HCMDx sets the string for the four high buffers (depending on x).

Example

Set the command buffer mode to digital input 1:

```
DIN1.MODE 9
```

Set the first sets of buffers to digital input 1:

```
DIN1.PARAM 1
```

Set the command DRV.OPMODE 0 to the high buffer:

```
DIN.HCMD1 DRV.OPMODE 1
```

Now, upon a rising edge in digital input 1, the drive mode is 1.

Related Topics

Command Buffer

1 Digital Inputs and Outputs

DIN1.MODE TO DIN7.MODE (→ p. 340), DIN1.PARAM TO DIN7.PARAM (→ p. 342), DIN.LCMD1 to DIN.LCMD4 (→ p. 335)

19.7.2 DIN.LCMD1 to DIN.LCMD4

| General Information | |
|---------------------|---|
| Type | NV parameter |
| Description | A buffer of commands to be used in digital input "command buffer" mode. |
| Units | N/A |
| Range | A string of up to 128 characters |
| Default Value | Empty |
| Data Type | String |
| Start Version | M_01-02-08-000 |

Description

DIN.LCMDx sets the string of commands to be used in the digital input mode command buffer. Digital input mode **9-Command buffer** can execute four different sets of command buffers.

Each set of command buffers contains two buffers:

- High buffer: Executes upon a rising edge of a digital input.
- Low buffer: Executes upon a falling edge of a digital input.

DIN.LCMDx sets the string for the four "low" buffers, depending on x.

Example

Set the command buffer mode to digital input 1:

```
DIN1.MODE 9
```

Set the first sets of buffers to digital input 1:

```
DIN1.PARAM 1
```

Set the command DRV.OPMODE 0 to the "low buffer":

```
DIN.LCMD1 DRV.OPMODE 0
```

Now, upon a falling edge in digital input 1, the drive mode is 0.

Related Topics

Command Buffer

1 Digital Inputs and Outputs

DIN1.MODE TO DIN7.MODE (→ p. 340), DIN1.PARAM TO DIN7.PARAM (→ p. 342), DIN.HCMD1 TO DIN.HCMD4 (→ p. 334)

19.7.3 DIN.ROTARY

| General Information | |
|---------------------|------------------------------|
| Type | R/O Parameter |
| Description | Reads the rotary knob value. |
| Units | N/A |
| Range | 0 to 99 |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3473h/1 | M_01-00-00-000 |

Description

DIN.ROTARY reads the rotary knob value.



Related Topics

- 1 Digital Inputs and Outputs

19.7.4 DIN.STATES

| General Information | |
|---------------------|---------------------------------|
| Type | R/O Parameter |
| Description | Reads the digital input states. |
| Units | N/A |
| Range | 0000000 to 1111111 |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-00-00-000 |

Description

DIN.STATES reads the states of the digital inputs. The leftmost bit represents digital input 1 (DIN1) and the rightmost bit represents digital input 7 (DIN7).

Related Topics

- 1 Digital Inputs and Outputs

19.7.5 DIN1.FILTER TO DIN7.FILTER

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Filter mode for digital inputs 1 to 71 to 3. |
| Units | N/A |
| Range | 0 to 3 |
| Default Value | 1 for DIN1 and DIN2 2 for DIN3 to DIN7 |
| Data Type | Integer |
| Start Version | M_01-03-07-000 |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|-------------|----------------------|
| EtherCAT COE and CANopen | 51CBh/0 | DIN1.FILTER | M_01-03-07-000 |
| | 51CCh/0 | DIN2.FILTER | |
| | 51CDh/0 | DIN3.FILTER | |
| | 51CEh/0 | DIN4.FILTER | |
| | 51CFh/0 | DIN5.FILTER | |
| | 51D0h/0 | DIN6.FILTER | |
| | 51D1h/0 | DIN7.FILTER | |

Description

This parameter sets the digital input filter configuration for channel x when followed with the values defined below. DINx.FILTER retrieves this information when not followed by data.

| Value | Description |
|------------------|---|
| DINX.FILTER 0 | The drive digital input channel detects all input signals with an input pulse width of ≥ 40 ns (no filtering applied). |
| DINX.FILTER 1 | The drive digital input channel detects all input signals with an input pulse width of ≥ 10.24 μ s, ± 0.64 μ s (fast filter applied). |
| DINX.FILTER 2 | The drive digital input channel detects all input signals with an input pulse width of ≥ 163 μ s, ± 10.24 μ s (standard filter applied). |
| DINX.FILTER 3 | The drive digital input channel detects all input signals with an input pulse width of ≥ 2.62 ms, ± 0.16384 ms (slow filter applied). |

Related Topics

Digital Inputs and Outputs

19.7.6 DIN1.INV to DIN7.INV

| General Information | |
|---------------------|--|
| Type | RW Parameter |
| Description | Sets the indicated polarity of a digital input mode. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Boolean |
| Start Version | M_01-00-00-000 |

Description

Sets the indicated polarity of a digital input mode.

Example

`DIN1.INV = 0` : Input is active high.

`DIN1.INV = 1` : Input is active low.

Related Topics

Digital Inputs and Outputs

19.7.7 DIN1.MODE TO DIN7.MODE

| General Information | |
|---------------------|-------------------------------|
| Type | R/W Parameter |
| Description | Sets the digital input modes. |
| Units | N/A |
| Range | 0 to 26 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|------------------------|----------------------|
| EtherCAT COE and CANopen | 3562h/0 | DIN1.MODE | M_01-00-00-000 |
| | 3565h/0 | DIN2.MODE | |
| | 3568h/0 | DIN3.MODE | |
| | 356Bh/0 | DIN4.MODE | |
| | 36F6h/0 | DIN5.MODE | |
| | 36F9h/0 | DIN6.MODE | |
| | 36FCh/0 | DIN7.MODE | |
| | 60FDh/0 | DIN1.MODE TO DIN6.MODE | |

Description

This parameter sets the functionality of the digital inputs. Digital inputs and corresponding X7 and X8 pin connectors are described in the *AKD PDMM Installation Manual*, section 8.16.4, Digital Inputs. The table below summarizes the digital input modes; for detailed descriptions of each mode, see Digital Inputs and Outputs (pg 1).

| DINx.MODE | Description | Task |
|-----------|--|------------|
| 0 | No function; off | None |
| 1 | Fault reset | Background |
| 2 | Start motion task (use DINx.PARAM for this task) | 1 KHz |
| 6 | Start jog | Background |
| 7 | Reserved | None |
| 8 | Zero latch | Background |
| 9 | Command buffer | Background |
| 10 | Control fault relay | Background |
| 11 | Home reference | 1 kHz |
| 12 | Reserved | None |
| 13 | Controlled Stop (→ p. 80) | 1 kHz |
| 14 | Reserved | None |
| 15 | Quick stop | Background |
| 17 | Activate electronic gear position shift | Background |

| DINx.MODE | Description | Task |
|-----------|---|------------|
| 18 | Positive limit switch | 4 kHz |
| 19 | Negative limit switch | 4kHz |
| 20 | Brake release | Background |
| 21 | Current limit | 4 kHz |
| 22 | Opmode and Command Source switch | Background |
| 23 | Change algebraic sign of the measured analog input voltage. | 1 kHz |
| 24 | Reserved | 1 kHz |
| 25 | Controlled stop without re-enable. | 1 kHz |
| 26 | Suspend Motion | 1 kHz |

Related Topics

Command Buffer

Digital Inputs (pg 1)

Controlled Stop (→ p. 80)

Clearing Faults (→ p. 199)

F245 (→ p. 183)

CS Parameters (→ p. 328)

19.7.8 DIN1.PARAM TO DIN7.PARAM

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets a value used as an extra parameter for digital inputs nodes. |
| Units | N/A |
| Range | -9,223,372,036,854,775,000 to +9,223,372,036,854,775,000 Note: Varies based on DINx.MODE. See below. |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version | |
|--------------------------|-----------------|----------------------|------------|
| EtherCAT COE and CANopen | DIN1.PARAM | M_01-00-00-000 | |
| | 3474h/2, 3474/9 | | DIN2.PARAM |
| | 3474h/3, 3474/A | | DIN3.PARAM |
| | 3474h/4, 3474/B | | DIN4.PARAM |
| | 3474h/5, 3474/C | | DIN5.PARAM |
| | 3474h/6, 3474/D | | DIN6.PARAM |
| | 3474h/7, 3474/E | | DIN7.PARAM |

Description

This parameter sets a value that is used as an extra parameter for digital inputs nodes.

Example

The digital input mode "Start motion task" is used to start a motion task. This mode uses an extra parameter as the ID of the motion task to be started.

Range

DINx.PARAM is used for various Digital Input modes. This causes the parameter's range to change based on the current Digital Input mode selected with the corresponding DINx.MODE

Listed below are the possible ranges for each Digital input mode.

If an input mode is not listed, then the default range above is used.

| Input Mode | Min | Max | Notes |
|------------|-----------------|-----------------|--|
| 2 | 0 | 128 | |
| 6 | Velocity Min | Velocity Max | This value changes based on user selected velocity units. |
| 9 | 0 | 4 | |
| 17 | Position Min | Position Max | This value changes based on user selected position units. |
| 21 | 0 | DRV.IPEAK | This value changes based on the specific AKD PDMMs drive limits. |

| Input Mode | Min | Max | Notes |
|------------|-----|------|--|
| 22 | 0 | 32 | See Digital Input Mode 22 for details. |
| 23 | 0 | 2056 | See Digital Input Mode 23 for details. |

Dependency on DINx.MODE

Typically, the user can set DINx.PARAM before the corresponding DINx.MODE is set. However, if DINx.PARAM is set before DINx.MODE is set, and the value of DINx.PARAM is outside the new DINx.MODE's range, then DINx.PARAM will be set to zero.

Example:

DIN1.MODE is set to 0 by default

DIN1.PARAM is set to 200

DIN1.MODE is changed to 2 (execute motion tasks)

200 is larger than the maximum for DIN1.MODE 2, so DIN1.PARAM will be set to 0 to prevent errors.

Related Topics

- 1 Digital Inputs and Outputs

19.7.9 DIN1.STATE TO DIN7.STATE

| General Information | |
|---------------------|---------------------------------------|
| Type | R/O Parameter |
| Description | Reads a specific digital input state. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

Description

Reads the state of one digital input according to the number identified in the command.

Related Topics

- 1 Digital Inputs and Outputs

19.7.10 DIN9.STATE to DIN11.STATE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Shows on selected pin if signal is high or low. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | U8 |
| Start Version | M_01-05-00-000 |

Description

This parameter allows the user to see the actual level of the input signal, when the IO is set to input mode. Parameter value is 0 if signal is low and 1 if signal is high. DIOx.INV can affect the value in this register.

This parameter can be read at any time. The value is only guaranteed to correspond to the output on the X9 connector when DRV.EMUEMODE is set to 10 and the DIOx.DIR is 0.

Related Topics

DRV.EMUEMODE (→ p. 392)

19.8 DIO Parameters

This section describes the DIO parameters.

| | |
|---|------------|
| 19.8.1 DIO9.INV to DIO11.INV | 347 |
| 19.8.2 DIO9.DIR to DIO11.DIR | 348 |

19.8.1 DIO9.INV to DIO11.INV

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Inverting the output voltage of the IO, when in the output direction. |
| Units | NA |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | U8 |
| Start Version | M_01-05-00-000 |

| Fieldbus | Index/Subindex | Parameter | Object Start Version |
|--------------------------|----------------|-----------|----------------------|
| EtherCAT COE and CANopen | 5258h/0 | DIO9.INV | M_01-05-00-000 |
| | 5254h/0 | DIO10.INV | |
| | 5256h/0 | DIO11.INV | |

Description

This parameter changes the logic sense of the differential input/output signals. When false, a logic 1 occurs when the + signal is higher than the – signal. When true, a logic 1 occurs when the – signal is higher than the + signal.

The drive output parameters DOUTx.STATE and DOUTx.STATEU are not affected by changes in this parameter. The drive input parameters DINx.STATE will be affected.

This parameter can be set at any time. It will be ignored unless DRV.EMUEMODE is set to 10.

Related Topics

DIN1.STATE TO DIN7.STATE (→ p. 344)

DOUT1.STATE AND DOUT2.STATE (→ p. 357)

DOUT1.STATEU AND DOUT2.STATEU (→ p. 358)

DRV.EMUEMODE (→ p. 392)

19.8.2 DIO9.DIR to DIO11.DIR

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Changing direction of the IOs from the X9 connector. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | U8 |
| Start Version | M_01-05-00-000 |

| Fieldbus | Index/Subindex | Parameter | Object Start Version |
|--------------------------|----------------|-----------|----------------------|
| EtherCAT COE and CANopen | 5257h/0 | DIO9.DIR | M_01-05-00-000 |
| | 5253h/0 | DIO10.DIR | |
| | 5255h/0 | DIO11.DIR | |

Description

This parameter changes the direction of the general purpose IO from the X9 connector. If DIOx.DIR is set 0 then the IO configured as an input, while if DIOx.DIR is 1 the IO is configured as an output.

DIO9.DIR controls pins 1 and 2

DIO10.DIR controls pins pin 4 and 5

DIO11.DIR controls pins pin 7 and 8.

This parameter can be set at any time. It will be ignored unless DRV.EMUEMODE is set to 10.

Related Topics

DIN1.STATE TO DIN7.STATE (→ p. 344)

DOUT1.STATE AND DOUT2.STATE (→ p. 357)

DOUT1.STATEU AND DOUT2.STATEU (→ p. 358)

DIO9.INV to DIO11.INV (→ p. 347)

DRV.EMUEMODE (→ p. 392)

19.9 DOUT Parameters

This section describes the DOUT parameters.

| | |
|--|------------|
| 19.9.1 DOUT.CTRL | 350 |
| 19.9.2 DOUT.RELAYMODE | 351 |
| 19.9.3 DOUT.STATES | 352 |
| 19.9.4 DOUT1.MODE and DOUT2.MODE | 353 |
| 19.9.5 DOUT1.PARAM AND DOUT2.PARAM | 355 |
| 19.9.6 DOUT1.STATE AND DOUT2.STATE | 357 |
| 19.9.7 DOUT1.STATEU AND DOUT2.STATEU | 358 |
| 19.9.8 DOUT9.MODE to DOUT11.MODE | 359 |
| 19.9.9 DOUT9.PARAM to DOUT11.MODE | 360 |
| 19.9.10 DOUT9.STATE to DOUT11.STATE | 361 |
| 19.9.11 DOUT9.STATEU to DOUT11.STATEU | 362 |

19.9.1 DOUT.CTRL

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the source of digital output(s) (firmware or fieldbus). |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Boolean |
| Start Version | M_01-00-00-000 |

Description

DOUT.CTRL sets the source of the digital output(s):

0 = Firmware controlled

1 = Fieldbus controlled

Related Topics

1.4 Digital Outputs

19.9.2 DOUT.RELAYMODE

| General Information | |
|---------------------|------------------------------|
| Type | R/W Parameter |
| Description | Indicates faults relay mode. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5060h/0 | M_01-00-00-000 |

Description

DOUT.RELAYMODE indicates the faults relay mode as follows:

If DOUT.RELAYMODE= 0 and faults exist, then the relay is open.

If DOUT.RELAYMODE= 0 and faults do not exist, then the relay is closed.

If DOUT.RELAYMODE = 1 and the drive is disabled, then the relay is open.

If DOUT.RELAYMODE = 1 and the drive is enabled, then the relay is closed.

Related Topics

1.4 Digital Outputs

19.9.3 DOUT.STATES

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the state of the digital output(s). |
| Units | N/A |
| Range | 0 to 11 |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-00-00-000 |

Description

DOUT.STATES reads the states of the digital output(s). The leftmost bit represents DOUT1 and the rightmost bit represents DOUT2.

Related Topics

1.4 Digital Outputs

19.9.4 DOUT1.MODE and DOUT2.MODE

| General Information | |
|---------------------|-------------------------------|
| Type | NV Parameter |
| Description | Sets the digital output mode. |
| Units | N/A |
| Range | 0 to 24 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-04-02-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 35AFh/0 | DOUT1.MODE |
| | 35B2h/0 | DOUT2.MODE |
| | | M_01-04-02-000 |

Description

DOUTx.MODE sets the functionality of the digital output(s). The table below summarizes the digital output modes; for detailed descriptions of each mode, see Digital Inputs and Outputs.

| DOUTx.MODE | Description |
|------------|----------------------------------|
| 0 | User (default = 0) |
| 1 | Mains ready |
| 2 | Software limit switch reached |
| 3 | Move complete |
| 4 | In position |
| 5 | Position greater than x |
| 6 | Position less than x |
| 7 | Drive produced warning |
| 8 | Drive enabled |
| 9 | Reserved |
| 10 | Motor brake |
| 11 | Drive produced fault |
| 12 | Absolute velocity greater than x |
| 13 | Absolute velocity less than x |
| 14 | Homing complete |
| 15 | Programmable Limit Switch Output |
| 16 | Command Buffer Active |
| 17 | Mt in Position |
| 20 | Controlled Stop Active |
| 21 | Immediate Fault Disable |
| 22 | In Rush Relay State |

| DOUTx.MODE | Description |
|------------|----------------|
| 23 | Compare Output |
| 24 | STO Status |

Related Topics

1.4 Digital Outputs

DOUT1.PARAM AND DOUT2.PARAM (→ p. 355)

19.9.5 DOUT1.PARAM AND DOUT2.PARAM

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets extra parameters for the digital output(s). |
| Units | N/A |
| Range | 0 Note: Range changes based on Digital Output Mode. See below. |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Name | Object Start Version |
|--------------------------|-------------------------------|-------------|----------------------|
| EtherCAT COE and CANopen | 3475h/1 (low), 3475h/3 (high) | DOUT1.PARAM | M_01-04-02-000 |
| | 3475h/2 (low), 3475h/4 (high) | DOUT2.PARAM | |

Description

Set the extra parameter needed for the digital output's calculations.

Range

DOUTx.PARAM is used for various Digital Output modes. This causes the parameter's range to change based on the current Digital Output mode selected with the corresponding DOUTx.MODE.

Below is a list of the possible range for each Digital Output Mode.

If an output mode is not listed, then the default range of 0 is used.

| Input Mode | Min | Max | Notes |
|------------|--------------|--------------|---|
| 3 | Position Min | Position Max | This value changes based on user selected position units. |
| 4 | Position Min | Position Max | This value changes based on user selected position units. |
| 5 | Position Min | Position Max | This value changes based on user selected position units. |
| 6 | Position Min | Position Max | This value changes based on user selected position units. |
| 12 | 0 | Velocity Max | This value changes based on user selected velocity units. |
| 13 | 0 | Velocity Max | This value changes based on user selected velocity units. |
| 15 | 0 | 255 | None. |
| 17 | Position Min | Position Max | This value changes based on user selected position units. |
| 23 | 0 | 1 | Selects the compare engine 0 (CMP0) or 1 (CMP1). |

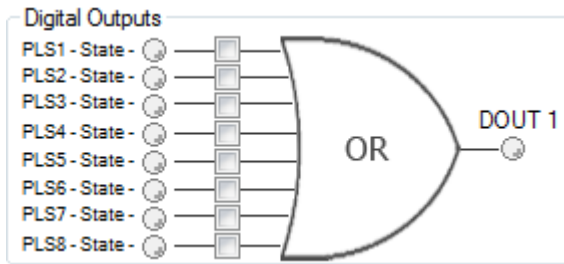
Dependency on DOUTx.MODE

Since the default range of DOUTx.PARAM does not allow a user to enter a value, DOUTx.MODE must be set to a mode which uses DOUTx.PARAM before a value can be set.

Each time DOUTx.MODE is changed, DOUTx.PARAM is automatically set to zero to prevent unintended interactions.

Digital Output Mode 15: PLS.STATE bits OR connected

The output mode produces a high signal if at least one of the PLS.STATE bits is high (the PLS is active) and if the corresponding bit in the DOUTx.PARAM parameter also has been set to high. The DOUTx.PARAM command connects the PLS.STATE bits to the digital output itself and thus acts as an enable mask. In mode 15 DOUTx.PARAM is set from the Digital Outputs section of the Programmable Limit Switches screen.



This mode is valid for all opmodes and command source combinations.

Example

```
|<- Bit 7 to 0 ->|
```

```
DOUT1.PARAM = 23 = 0b 0 0 0 1 0 1 1 1 (Binary code)
```

The digital output 1 is active when bit 0 or bit 1 or bit 2 or bit 4 of PLS.STATE is high. All other bits within PLS.STATE are not considered by the digital output mode due to the DOUT1.PARAM setting. Do not use decimal places for the DOUTx.PARAM parameter for this particular digital output mode.

Related Topics

1.4 Digital Outputs

19.9.6 DOUT1.STATE AND DOUT2.STATE

| General Information | |
|---------------------|---------------------------------|
| Type | R/O Parameter |
| Description | Reads the digital output state. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

Description

Reads the state of one digital output according to the value stated in the command.

Related Topics

1.4 Digital Outputs

19.9.7 DOUT1.STATEU AND DOUT2.STATEU

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the state of the digital output node. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-01-01-000 |

Description

Set the state of the digital output node as follows:

0 = deactivated

1 = activated

DOUT1.STATEU and DOUT2.STATEU are used when [DOUTx.MODE](#) = 0 (user mode).

Related Topics

1.4 Digital Outputs

19.9.8 DOUT9.MODE to DOUT11.MODE

| General Information | |
|---------------------|-------------------------------|
| Type | NV Parameter |
| Description | Sets the digital output mode. |
| Units | N/A |
| Range | 0, 23 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | |

Description

DOUTx.MODE sets the functionality of the digital output(s). The table below summarizes the digital output modes; for detailed descriptions of each mode, see Digital Inputs and Outputs.

| DOUTx.MODE | Description |
|------------|--------------------|
| 0 | User (default = 0) |
| 23 | Compare Output |

Related Topics

1.4 Digital Outputs

DOUT1.PARAM AND DOUT2.PARAM (→ p. 355)

Compare Engines (→ p. 75)

19.9.9 DOUT9.PARAM to DOUT11.MODE

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Selects Compare Engine 0 or 1 for DOUTx. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | N/A |

Description

DOUT9.PARAM, DOUT10.PARAM, and DOUT11.PARAM select the compare engine (CMP0 or CMP1) for the output.

| DOUTx.PARAM value | Description |
|-------------------|------------------|
| 0 | Compare Engine 0 |
| 1 | Compare Engine 1 |

Related Topics

1.4 Digital Outputs

19.9.10 DOUT9.STATE to DOUT11.STATE

| General Information | |
|---------------------|---|
| Type | NV parameter |
| Description | Shows on selected pin if signal is high or low. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | U8 |
| Start Version | M_01-05-00-000 |

Description

This parameter allows the user to see the actual level of the output signal, when the IO is set to output mode. Parameter value is 0 if signal is low and 1 if signal is high. DIOx.INV can affect the signals driven onto the X9 connector.

This parameter can be read at any time. The value is only guaranteed to correspond to the output on the X9 connector when DRV.EMUEMODE is set to 10 and the DIOx.DIR is 0.

Related Topics

DOUT9.STATEU to DOUT11.STATEU (→ p. 362)

DRV.EMUEMODE (→ p. 392)

19.9.11 DOUT9.STATEU to DOUT11.STATEU

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Allows user to set level of selected pin to high or low. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | U8 |
| Start Version | M_01-05-00-000 |

Description

This parameter allows the user to set the level of the output signal, when the IO is set to output mode. Parameter value is 0 if signal is low and 1 if signal is high. DIOx.INV can affect the signals driven onto the X9 connector.

This parameter can be written at any time. The value is only guaranteed to correspond to the output on the X9 connector when DRV.EMUEMODE is set to 10 and the DIOX.DIR is 0.

Example

The following settings set the direction for the differential signals on pin 4 and 5, so that the output will have a high level signal.

First set the following settings:

```
DRV.EMUEMODE 10
DIO10.DIR 1
DOUT10.STATEU 1
```

Then change the level of the signal:

```
DOUT.STATEU 0
```

or

```
DIO10.INV
```

Note: Inverting the signal will also alter the signal in input mode.

Related Topics

DRV.EMUEMODE (→ p. 392)

19.10 DRV Parameters

This section describes the DRV parameters.

| | |
|-------------------------------------|-----|
| 19.10.1 DRV.ACC | 366 |
| 19.10.2 DRV.ACTIVE | 368 |
| 19.10.3 DRV.BLINKDISPLAY | 369 |
| 19.10.4 DRV.BOOTTIME | 370 |
| 19.10.5 DRV.CLRCRASHDUMP | 371 |
| 19.10.6 DRV.CLRFAULTHIST | 372 |
| 19.10.7 DRV.CLRFAULTS | 373 |
| 19.10.8 DRV.CMDDELAY | 374 |
| 19.10.9 DRV.CMDSOURCE | 375 |
| 19.10.10 DRV.CRASHDUMP | 376 |
| 19.10.11 DRV.CUSTOMIDENTIFIER | 377 |
| 19.10.12 DRV.DBILIMIT | 378 |
| 19.10.13 DRV.DEC | 379 |
| 19.10.14 DRV.DIFVAR | 381 |
| 19.10.15 DRV.DIR | 382 |
| 19.10.16 DRV.DIS | 384 |
| 19.10.17 DRV.DISMODE | 385 |
| 19.10.18 DRV.DISSOURCES | 386 |
| 19.10.19 DRV.DISSOURCESMASK | 387 |
| 19.10.20 DRV.DISTO | 388 |
| 19.10.21 DRV.DOWNLOADALLOWED | 389 |
| 19.10.22 DRV.EMUHECKSPEED | 390 |
| 19.10.23 DRV.EMUEDIR | 391 |
| 19.10.24 DRV.EMUEMODE | 392 |
| 19.10.25 DRV.EMUEMTURN | 394 |
| 19.10.26 DRV.EMUEPULSEWIDTH | 395 |
| 19.10.27 DRV.EMUERES | 396 |
| 19.10.28 DRV.EMUESTEPCMD | 397 |
| 19.10.29 DRV.EMUESTEPCMDPIN | 398 |
| 19.10.30 DRV.EMUESTEPCMDPOUT | 399 |
| 19.10.31 DRV.EMUEZOFFSET | 400 |
| 19.10.32 DRV.EN | 401 |

| | |
|--|-----|
| 19.10.33 DRV.ENDEFAULT | 402 |
| 19.10.34 DRV.ERRORLIST | 403 |
| 19.10.35 DRV.FAULTDISPLAYMODE | 404 |
| 19.10.36 DRV.FAULTHIST | 405 |
| 19.10.37 DRV.FAULT1 to DRV.FAULT10 | 406 |
| 19.10.38 DRV.FAULTS | 407 |
| 19.10.39 DRV.HANDWHEELSRC | 408 |
| 19.10.40 DRV.HELP | 409 |
| 19.10.41 DRV.HELPALL | 410 |
| 19.10.42 DRV.HWENABLE | 411 |
| 19.10.43 DRV.HWENDELAY | 412 |
| 19.10.44 DRV.HWENMODE | 413 |
| 19.10.45 DRV.ICONT | 414 |
| 19.10.46 DRV.INFO | 415 |
| 19.10.47 DRV.IPEAK | 417 |
| 19.10.48 DRV.IZERO | 418 |
| 19.10.49 DRV.LIST | 419 |
| 19.10.50 DRV.LOGICVOLTS | 420 |
| 19.10.51 DRV.MEMADDR | 421 |
| 19.10.52 DRV.MEMDATA | 422 |
| 19.10.53 DRV.MOTIONDISSOURCES | 423 |
| 19.10.54 DRV.NAME | 424 |
| 19.10.55 DRV.NVCHECK | 425 |
| 19.10.56 DRV.NVLIST | 426 |
| 19.10.57 DRV.NVLOAD | 427 |
| 19.10.58 DRV.NVSAVE | 428 |
| 19.10.59 DRV.ONTIME | 429 |
| 19.10.60 DRV.OPMODE | 430 |
| 19.10.61 DRV.READFORMAT | 431 |
| 19.10.62 DRV.REBOOT | 432 |
| 19.10.63 DRV.RSTVAR | 433 |
| 19.10.64 DRV.RUNTIME | 434 |
| 19.10.65 DRV.SETUPREQBITS | 435 |
| 19.10.66 DRV.SETUPREQLIST | 437 |

| | |
|--|------------|
| 19.10.67 DRV.STOP | 438 |
| 19.10.68 DRV.TEMPERATURES | 439 |
| 19.10.69 DRV.TYPE | 440 |
| 19.10.70 DRV.VER | 441 |
| 19.10.71 DRV.VERIMAGE | 442 |
| 19.10.72 DRV.WARNING1 to DRV.WARNING3 | 443 |
| 19.10.73 DRV.WARNINGS | 444 |
| 19.10.74 DRV.ZERO | 445 |

19.10.1 DRV.ACC

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Describes the acceleration ramp for the velocity loop. |
| Units | Depends on UNIT.ACCROTARY (pg 1) or UNIT.ACCLINEAR Rotary: rps/s, rpm/s, deg/s ² , (custom units)/s ² , rad/s ² Linear: counts/s ² , mm/s ² , μm/s ² , (custom units)/s ² |
| Range | Note: The range and default values of (custom units)/s ² units depend on the values of PIN and POUT. The range and default values listed in this table are derived from the default values of PIN and POUT. Rotary: 0.002 to 833,333.333 rps/s 0.112 to 50,000,000.000 rpm/s 0.009 to 300,000,000.000 deg/s ² 0.155 to 4,166,666.752 (custom units)/s ² 0.012 to 5,235,987.968 rad/s ² Linear: 16,000.000 to 3,579,139,408,000.000 counts/s ² 0.031*MOTOR.PITCH (→ p. 617) to 833,333.333*MOTOR.PITCH (→ p. 617) mm/s ² 30.995*MOTOR.PITCH (→ p. 617) to 2,147,483.647*MOTOR.PITCH (→ p. 617) μm/s ² 0.155 to 2,147,483.647 (custom units)/s ² |
| Default Value | Note: The range and default values of (custom units)/s ² units depend on the values of PIN and POUT. The range and default values listed in this table are derived from the default values of PIN and POUT. Rotary: 166.669 rps/s 10,000.000 rpm/s 60,000.000 deg/s ² 833.333 (custom units)/s ² 1,047.2 rad/s ² Linear: 715,840,000.000 counts/s ² 166.714*MOTOR.PITCH (→ p. 617) mm/s ² 166,714.191*MOTOR.PITCH (→ p. 617) μm/s ² 833.571 (custom units)/s ² |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|-------------------------------|----------------------|
| EtherCAT COE and CANopen | 3501h/0 6083h (→ p. 996)/0 | M_01-00-00-000 |

Description

Describes the acceleration ramp for the velocity central loop.

Related Topics

10.1 Limits

DRV.DEC (→ p. 379)

19.10.2 DRV.ACTIVE

| General Information | |
|---------------------|-------------------------------------|
| Type | R/O Parameter |
| Description | Reads the enable status of an axis. |
| Units | N/A |
| Range | 0, 1, 3 |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

Description

DRV.ACTIVE reads the enable status of an axis as follows:

- DRV.ACTIVE = 0 drive disabled
- DRV.ACTIVE = 1 drive enabled
- DRV.ACTIVE = 3 drive enabled and in dynamic brake mode

There is no state 2.

When the drive is in state 3, the drive display shows a blinking decimal point.

If an axis is not enabled (DRV.ACTIVE is 0), but DRV.EN (→ p. 401) is 1 and the hardware enable is high, read the value of DRV.DISSOURCES (→ p. 386) to query the reason that the drive is not enabled.

Related Topics

10.3 Enable/Disable

DRV.EN (→ p. 401), DRV.DISSOURCES (→ p. 386)

19.10.3 DRV.BLINKDISPLAY

| General Information | |
|---------------------|---|
| Type | Command |
| Description | Causes the display to blink for 10 seconds. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

DRV.BLINKDISPLAY causes the drive display located on the front of the drive to blink for 10 seconds.

This command allows the user to identify the drive that is currently communicating with EWV¹.

¹Embedded Workbench Views

19.10.4 DRV.BOOTTIME

| General Information | |
|---------------------|--|
| Type | R/O |
| Description | Returns the time when the current session booted up. |
| Units | Days:Hours:Minutes:Seconds |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-06-05-000 |

Description

DRV.BOOTTIME returns the time that the current session started running. This time includes total amount of time from all previous sessions.

This keyword can be used with DRV.RUNTIME to determine the length of time the drive has been running since it was last power cycled.

Session Time = DRV.RUNTIME – DRV.BOOTTIME

Related Topics

DRV.RUNTIME (→ p. 434)

19.10.5 DRV.CLRCRASHDUMP

| General Information | |
|---------------------|------------------------------------|
| Type | Command |
| Description | Clears old diagnostic information. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-04-00-005 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

Since the NV memory block is overwritten, but never erased, the DRV.CRASHDUMP command always shows the diagnostic information for the most recent crash. In some instances it is helpful to erase an old crashdump. Use DRV.CLRCRASHDUMP for this purpose.

19.10.6 DRV.CLRFAULTHIST

| General Information | |
|---------------------|---|
| Type | Command |
| Description | Clears the fault history log in the NV. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

DRV.CLRFAULTHIST clears the fault history from the nonvolatile memory of the drive.

This command erases all faults returned by DRV.FAULTHIST (→ p. 405).

19.10.7 DRV.CLRFAULTS

| General Information | |
|---------------------|--|
| Type | Command |
| Description | Tries to clear all active faults in the drive. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

When DRV.CLRFAULTS is sent, the drive will try to clear all active faults. When a fault occurs, the fault is registered in the drive fault handler. DRV.CLRFAULTS clears the fault from the drive fault handler. However, if the fault still exists in the system, DRV.CLRFAULTS fails and the fault is re-registered in the fault handler.

If the DRV.CLRFAULTS succeeds, then the reply to DRV.FAULTS states that no faults exist. If the condition that triggered the fault is still present, the fault condition will remain.

See Fault and Warning Messages (→ p. 177) for details regarding the behavior of individual faults.

Note that executing a drive disable (DRV.DIS (→ p. 384)) followed by a drive enable (DRV.EN (→ p. 401)) has the same effect as executing DRV.CLRFAULTS.

Related Topics

Clearing Faults (→ p. 199)

DRV.FAULTS (→ p. 407), DRV.EN (→ p. 401), DRV.DIS (→ p. 384)

19.10.8 DRV.CMDDELAY

| General Information | |
|---------------------|---|
| Type | Command |
| Description | Issues a delay before next command is executed. |
| Units | ms |
| Range | 0 to 5,000 ms |
| Default Value | 0 ms |
| Data Type | Float |
| Start Version | M_01-03-00-000 |

Description

This parameter is used when drive commands are used in a script and a delay is needed between the execution of two consecutive commands. DRV.CMDDELAY creates a delay in the execution of drive commands. In the period of time specified, no commands are executed. This feature is especially useful for command buffers.

Example

If the script is:

```
DRV.EN
IL.CMDU 0.1
```

then DRV.CMDDELAY is used between the two entries to delay execution 5 ms until the drive is enabled:

```
DRV.EN
DRV.CMDDELAY 5
IL.CMDU 0.1
```

19.10.9 DRV.CMDSOURCE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the command source (service, fieldbus, analog input, gearing, digital, or Bode). |
| Units | N/A |
| Range | 0 to 5 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

Description

DRV.CMDSOURCE specifies the source of the command to the drive. DRV.OPMODE (→ p. 430) sets the operation mode to the relevant control loop.

DRV.CMDSOURCE values can be set as follows:

| Value | Description |
|-------|-------------------------|
| 0 | Service, TCP/IP command |
| 1 | Fieldbus command |
| 2 | Gearing command |
| 3 | Analog command |
| 5 | Program command |

If DRV.CMDSOURCE is set to 5 then DRV.OPMODE must be set to 3.

DRV.CMDSOURCE can be changed while the drive is enabled or disabled. If you use the terminal to change the operation mode, then it is recommended that you disable the drive before changing the command source.

WARNING

If you change DRV.CMDSOURCE from the terminal while the drive is enabled, the system may experience a step change in command.

Example

To set the command source to the TCP/IP channel and the operation mode to velocity:

```
-->DRV.CMDSOURCE 0
```

```
-->DRV.OPMODE 1
```

Related Topics

DRV.OPMODE (→ p. 430)

19.10.10 DRV.CRASHDUMP

| General Information | |
|---------------------|---|
| Type | Command |
| Description | Retrieves diagnostic information after the drive crashes. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

Description

Drives rarely crash, but if a crash occurs, information that can help diagnose the cause of a crash is saved to the nonvolatile (NV) memory within the drive. After the drive is restarted, you can use the DRV.CRASHDUMP command to retrieve this diagnostic information, which can be emailed to Kollmorgen for further support.

If the drive crashes (display flashes an F and three bars), it saves the diagnostic information to a specific block of the drive NV memory. The DRV.CRASHDUMP command then prints the diagnostic information from this NV memory block. Subsequent crash conditions will overwrite the NV memory block. Since the NV memory block is overwritten, but never erased, the DRV.CRASHDUMP command always shows the diagnostic information for the most recent crash.

19.10.11 DRV.CUSTOMIDENTIFIER

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the customer defined identifier string for a device. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-12-05-000 |

Fieldbus Information

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 2032h/0 | M_01-12-05-000 |

Description

Use DRV.CUSTOMIDENTIFIER to assign a unique string to a device. The string must use the following formatting rules:

- ASCII characters only
- 32 character max length
- No spaces in string

The string is displayed on the device overview screen and in the start page's device discovery table.

WorkBench has found the following devices.

[Is your device is not shown?](#)

| Name | Status | IP Address | MAC Address | Model Number | Firmware Version | Custom Identifier |
|----------|--------|---------------|--------------|----------------------|-----------------------------|--------------------|
| \^<>^/ | Free | 10.154.95.119 | 00240B007448 | AKD-P00306-NBEC-0000 | 01-12-04-000-58763-alpha | - |
| AKD_63 | Busy | 10.154.95.63 | 00240B010C4D | AKD-P00306-NBEC-0000 | 01-12-04-000-00000_2014-... | MyCustomIdentifier |
| MAI AM-7 | Free | 10.154.94.119 | 00221000E5A5 | AKD-P00306-NBEC-0000 | M_01-12-04-000 | |

Related Topics

DRV.NAME (→ p. 424)

19.10.12 DRV.DBILIMIT

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the maximum amplitude of the current for dynamic braking. |
| Units | Arms |
| Range | 0 to minimum of drive peak current (DRV.IPEAK) and motor peak current (MOTOR.IPEAK). |
| Default Value | Minimum of drive continuous current (DRV.ICONT) and motor continuous current (MOTOR.ICONT). |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3444h/0 | M_01-00-00-000 |

Description

This parameter sets the maximum amplitude of the current for dynamic braking.

Example

Setting DRV.DBILIMIT to 2 limits the dynamic brake current to 2 Arms.

Related Topics

10.4 Controlled Stop

19.6 CS Parameters

10.5 Dynamic Braking

DRV.DISMODE (→ p. 385)

19.10.13 DRV.DEC

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the deceleration value for the velocity loop. |
| Units | Depends on UNIT.ACCROTARY (pg 1) or UNIT.ACCLINEAR (pg 1) Rotary: rps/s, rpm/s, deg/s ² , (custom units)/s ² , rad/s ² Linear: counts/s ² , mm/s ² , μm/s ² , (custom units)/s ² |
| Range | Rotary: 0.002 to 833,333.333 rps/s 0.112 to 50,000,000.000 rpm/s 0.009 to 300,000,000.000 deg/s ² 0.155 to 4,166,666.752 (custom units)/s ² 0.012 to 5,235,987.968 rad/s ² Linear: 16,000.000 to 3,579,139,408,000.000 counts/s ² 0.031*MOTOR.PITCH to 833,333.333*MOTOR.PITCH mm/s ² 30.994*MOTOR.PITCH to 833,333,333.333*MOTOR.PITCH μm/s ² 0.155 to 4,166,666.667 (custom units)/s ² |
| Default Value | Rotary: 166.669 rps/s 10,000.000 rpm/s 60,000.000 deg/s ² 833.333 (custom units)/s ² 1,047.2 rad/s ² Linear: 715,840,000.000 counts/s ² 166.71*MOTOR.PITCH4MOTOR.PITCH (→ p. 617) mm/s ² 166,714.191*MOTOR.PITCHMOTOR.PITCH (→ p. 617) μm/s ² 833.571 (custom units)/s ² |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|-------------------------------|----------------------|
| EtherCAT COE and CANopen | 3522h/0 6084h (→ p. 996)/0 | M_01-00-00-000 |

Description

DRV.DEC sets the deceleration value for the velocity loop command (VL.CMDU (→ p. 730)) and for the analog

velocity command (). The operation mode (DRV.OPMODE (→ p. 430)) must be set to velocity mode for this command to function.

Related Topics

10.4 Controlled Stop

10.1 Limits

DRV.ACC (→ p. 366), DRV.OPMODE (→ p. 430)

19.10.14 DRV.DIFVAR

| General Information | |
|---------------------|---|
| Type | R/O |
| Description | Lists all parameters which differ from their default value. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-05-01-000 |

Description

This parameter displays all parameters which have different values than their default setting. The actual value of each parameter is directly shown after the command name and the corresponding default value is then shown in brackets.

This command also shows differences in parameters which hold a string, such as DRV.NAME.

Example

```
-->DRV.DIFVAR
DRV.EMUEMODE 10 (0)
DRV.NAME MyDrive(no-name)
FB1.ENCREAS 0(1024)
IL.KP 50.009(24.811)
PL.KP 99.998(49.999)
VL.KP 0.108(0.000)
```

19.10.15 DRV.DIR

| General Information | |
|---------------------|--------------------------|
| Type | R/W Parameter |
| Description | Changes drive direction. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

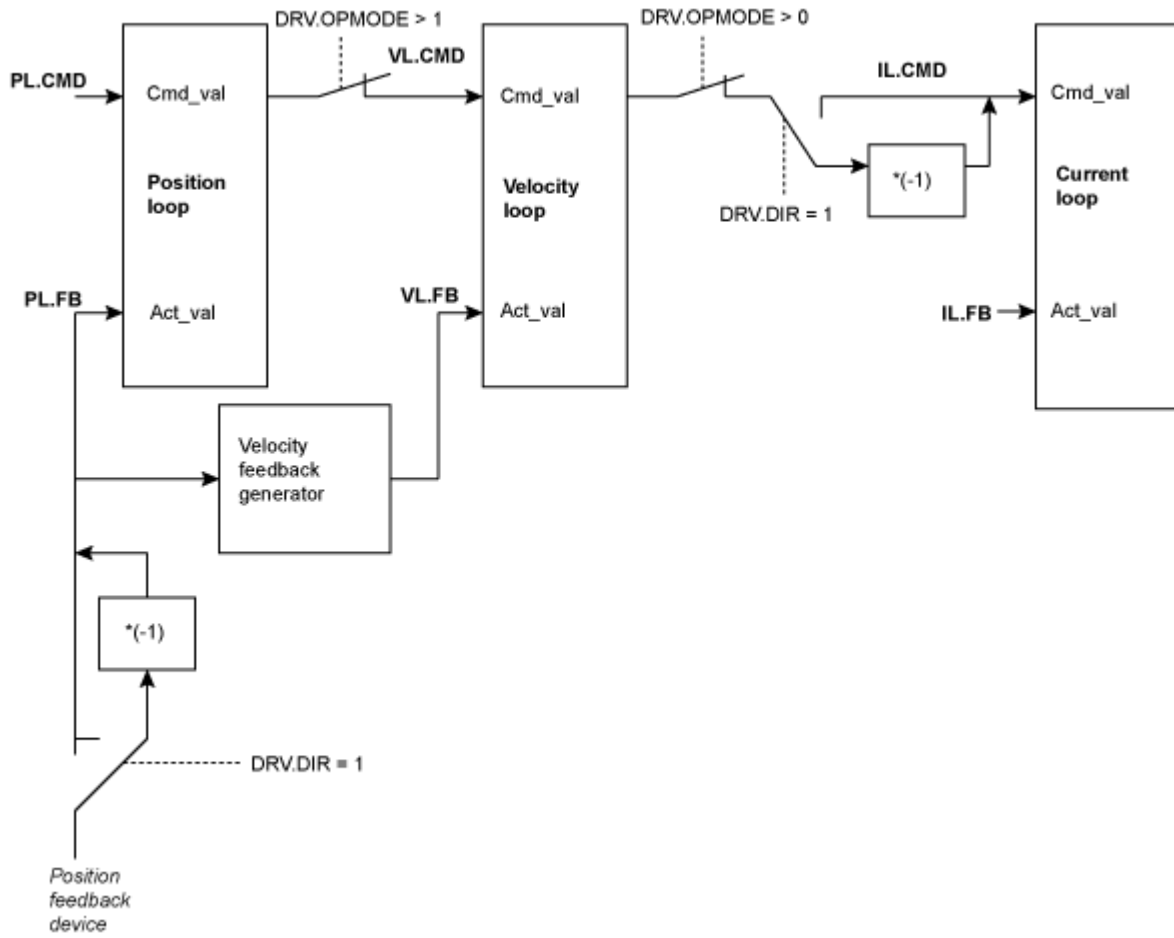
| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 352Ah/0 | M_01-00-00-000 |

Description

DRV.DIR changes the direction of the motor by changing the algebraic sign of the current command and position feedback value according to the figure below.

Note the following when using DRV.DIR:

- You can only change the DRV.DIR command when the drive is disabled.
- The drive status changes to "Axis not homed" as soon as the DRV.DIR parameter changes value (see DRV.MOTIONSTAT).
- You must verify the settings of the hardware limit switches. If necessary, switch the positive and negative hardware limit switches by swapping the wires at the digital inputs.



19.10.16 DRV.DIS

| General Information | |
|---------------------|--|
| Type | Command |
| Description | Disables the axis (software). |
| Units | N/A |
| Range | N/A |
| Default Value | Analog drive software enabled. All other types of drive software disabled. |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3443h/0 | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

DRV.DIS issues a software disable to the drive. The method by which the drive will be disabled (either immediately or with a ramp down first) is controlled by DRV.DISMODE (→ p. 385).

By querying the value of DRV.ACTIVE (→ p. 368), you can check whether the drive is currently enabled or disabled.

By querying the value of DRV.DISSOURCES (→ p. 386), you can check whether the software enable bit is high (software enabled was issued by executing DRV.EN) or the software enable bit is low (software disable was issued by executing DRV.DIS).

If DRV.DIS is commanded the emergency timeout is started. If the drive does not disable or activate dynamic brake within DRV.DISTO (→ p. 388), fault "F703" (→ p. 193) is reported.

Related Topics

Enable/Disable (→ p. 77)

15.4 Clearing Faults

10.4 Controlled Stop

DRV.EN (→ p. 401), DRV.DISSOURCES (→ p. 386), DRV.ACTIVE (→ p. 368), DRV.DISMODE (→ p. 385), DRV.DISTO (→ p. 388)

19.10.17 DRV.DISMODE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Selects among disable immediately or stop and then disable options. |
| Units | N/A |
| Range | 0 to 3 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 35FFh/0 | M_01-00-00-000 |

Description

DRV.DISMODE sets the drive reaction to a DRV.DIS (→ p. 384) (software disable) or fieldbus command.

NOTE

You must disable the drive in order to set DRV.DISMODE.

| Value | Behavior |
|-------|---|
| 0 | Disable axis immediately. If MOTOR.BRAKEIMM = 1, the brake is applied as soon as the power stage disables. |
| 1 | Use dynamic brake to ramp down. The drive remains in the dynamic brake state after the motor has stopped. The drive is disabled in the sense that it does not close the control loop and cannot perform a motion, but PWM stays active. |
| 2 | Use a controlled stop to ramp down and then disable the drive. |
| 3 | Use a controlled stop to ramp down, and then use dynamic brake. The drive remains in the dynamic brake state after the motor has stopped. The drive is disabled in the sense that it does not close the control loop and cannot perform a motion, but PWM stays active. |

⚠ WARNING

Be careful with vertical loads when modifying this parameter. Coordinate this parameter's correct setting properly with the drive brake settings. If these settings are not coordinated, then vertical loads may have no stopping or holding force when the drive is disabled and the load could fall. See Using AKD PDMM in a Vertical Axis (→ p. 68).

Related Topics

Enable/Disable (→ p. 77)

10.4 Controlled Stop

19.6 CS Parameters

10.5 Dynamic Braking

DRV.DBILIMIT (→ p. 378), DRV.DISTO (→ p. 388), CS.VTHRESH (→ p. 332)

19.10.18 DRV.DISSOURCES

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Returns the possible reason for a drive disable. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3443h/0 | M_01-00-00-000 |

Description

DRV.DISSOURCES is a bitwise parameter that returns the status of possible causes of a drive disable. If this parameter is 0, then the drive is enabled.

The return value specific bits are as follows:

| Bit | Status and Response | Drive Action |
|-----|--|--|
| 0 | Software disable (execute DRV.EN to issue software enable) | Performs action configured by DRV.DISMODE. |
| 1 | Fault exists (read DRV.FAULTS to get the active faults) | Device disables using the defined reaction for the detected fault. |
| 2 | Hardware disable (remote enable input is low) | Power stage disabled. |
| 3 | In-rush disable (the in-rush relay is opened) | Performs action configured by DRV.DISMODE. |
| 4 | Initialization disable (the drive did not finish the initialization process) | Power stage disabled. |
| 5 | Controlled stop disable from a digital input. | Controlled stop. |

Related Topics

Controlled Stop (→ p. 80)

DRV.ACTIVE, DRV.FAULTS, DRV.EN, DRV.DIS

19.10.19 DRV.DISSOURCESMASK

| General Information | |
|---------------------|---|
| Type | R/O parameter |
| Description | Returns a bitmap of all bits in DRV.DISSOURCES that can be set to one on a given drive. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-07-00-000 |

Description

DRV.DISSOURCESMASK is a bitwise parameter that returns all possible causes of a drive disable. If a bit is set, the drive can be disabled for this reason.

The return value specific bits are as follows:

| Bit | Status and Response |
|-----|--|
| 0 | Software triggered disable |
| 1 | Fault exists |
| 2 | Hardware Enable input inactive |
| 3 | In-rush relay state prevents enable |
| 4 | Initialization not complete |
| 5 | Controlled stop requested from digital input |
| 6 | Field bus requested disable |

For most AKD PDMM models, this parameter will return the value 63.

Related Topics

DRV.DISSOURCES (→ p. 386)

19.10.20 DRV.DISTO

| General Information | |
|---------------------|----------------------------|
| Type | R/W Parameter |
| Description | Sets the emergency timeout |
| Units | ms |
| Range | 0 to 120,000 ms |
| Default Value | 1,000 ms |
| Data Type | U32 |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3445h/0 | M_01-00-00-000 |

Description

This timer starts when DRV.DIS (→ p. 384) is issued (regardless of the DRV.DIS (→ p. 384) origin). After this timeout elapses, the actual state of the drive is compared to the DRV.DISMODE (→ p. 385) setting. If the actual state does not match the DRV.DISMODE (→ p. 385) setting, a fault is reported and the hardware immediately executes the DRV.DISMODE setting (for instance, disable or activate dynamic brake). Setting DRV.DISTO to 0 will disable the timeout.

Related Topics

10.4 Controlled Stop

DRV.DIS (→ p. 384), DRV.DISMODE (→ p. 385)

19.10.21 DRV.DOWNLOADALLOWED

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Returns whether the drive is in a safe state to begin a firmware download. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-09-02-000 |

Description

This parameter returns whether the drive is in a safe state to begin a firmware download. A download can only be performed if the value is one. If the value is zero, EWV¹ will block the download.

For an AKD PDMM or NSD, the value depends on the drive enable status. For a CPS, the value depends on the enable states of the NSDs on its string. If one more more NSDs are enabled on a string, download to the CPS will be blocked.

Related Topics

10.1 Limits

¹Embedded Workbench Views

19.10.22 DRV.EMUECHECKSPEED

| General Information | |
|---------------------|--|
| Type | R/W |
| Description | Enable/disable comparison of motor velocity and maximum emulated encoder velocity. See Fault F486 for details. |
| Units | None |
| Range | 0 to 1 |
| Default Value | 0 (disabled) |
| Data Type | Boolean |
| Start Version | M_01-06-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 520Eh/0 | M_01-06-00-000 |

Description

Enables comparison of the current motor velocity (or EEO pulse rate for DRV.EMUEMODE (→ p. 392) 15 to 17) against the maximum speed the emulated encoder output can generate. If the commanded speed exceeds DRV.EMUECHECKSPEED, fault F486 is raised.

For DRV.EMUEMODE 1 to 14, the commanded speed is based on lines/rev (DRV.EMUERES (→ p. 396)) and the pulse width (DRV.EMUEPULSEWIDTH (→ p. 395)).

For DRV.EMUEMODE 15 to 17 the commanded speed depends on the rate defined by changes in DRV.EMUESTEPCMD.

Related Topics

DRV.EMUEMODE

19.10.23 DRV.EMUEDIR

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the direction of the emulated encoder output (EEO) signal. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3493h/0 | M_01-00-00-000 |

Description

This parameter sets the direction of the emulated encoder output (EEO). DRV.DIR (→ p. 382) also effects the output direction (through an XOR, "exclusive or", operation). The drive uses DRV.DIR (→ p. 382) and DRV.EMUEDIR to decide the direction of the EEO. If DRV.DIR (→ p. 382) and DRV.EMUEDIR have the same value, then DRV.EMUEDIR is set to 0 (meaning an increase in the motor feedback will result an increase of the EEO and vice-versa). If these parameters have different values, then DRV.EMUEDIR is set to 1 (meaning an increase in the motor feedback will result in a decrease of the EEO and vice-versa).

19.10.24 DRV.EMUEMODE

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the mode of the emulated encoder output (EEO) connector. |
| Units | N/A |
| Range | 0 to 17 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3534h/0 | M_01-00-00-000 |

Description

When the emulated encoder output (EEO) is configured to generate an index pulse (Z pulse) once per rev (DRV.EMUEMODE is 1, 6, or 8) this parameter and DRV.EMUEZOFFSET define the location of the Z pulse. DRV.EMUEZOFFSET defines the position of the Z pulse within one revolution. The pulse is generated each time this position is reached. For example, if this value is set to 32,768 (180 degrees), the Z pulse will be generated at 180 degrees, 540 degrees, 900 degrees, and so on.

When the EEO is configured to generate an absolute index pulse (DRV.EMUEMODE is 2, 7, or 9) this parameter, DRV.,EMUEZOFFSET and DRV.EMUETURN define the location of the Z pulse. DRV.EMUETURN is used to define which turn of the position range the Z pulse is located. DRV.EMUEZOFFSET is used to define the position of the Z pulse within one revolution. For example, if DRV.EMUETURN is set to 1, and DRV.EMUEZOFFSET is set to 32,768 (180 degrees), then the index pulse will be generated at position 540 degrees.

When the EEO is configured in pass through mode (DRV.EMUEMODE is 12, 13, or 14) the EEO Z pulse will be generated each time the encoder's physical Z pulse is generated (the physical Z pulse is "passed through").

DRV.EMUEMODE sets the EEO connector to act as either an input or output as follows.

| Setting | Function |
|-----------------|--|
| 0 (recommended) | Input (see to select the type of inputs the secondary feedback will accept) |
| 1 | EEO Output, A/B with once per rev index |
| 2 | EEO Output, A/B with absolute index pulse. |
| 3 | Input, A/B signals (Deprecated) |
| 4 | Input, step and direction signals (Deprecated) |
| 5 | Input, up/down Signals (Deprecated) |
| 6 | Step/Dir with one Z-pulse/rev |
| 7 | Step/Dir with one absolute Z-pulse (depends on DRV.EMUEOFFSET and DRV.EMUETURN) |
| 8 | CW/CCW output with one Z-pulse/rev |
| 9 | CW/CCW output with one absolute Z-pulse (depends on DRV.EMUEOFFSET and DRV.EMUETURN) |

| Setting | Function |
|---------|---|
| 10 | Allows the X9 connector to be used as a General Purpose I/O or SynqNet fieldbus controlled I/O |
| 11 | FB3 Input (Tertiary feedback is reported with FB3.P (→ p. 507)). Use FB3.MODE (→ p. 504) to select the feedback type. |
| 12 | Output - A/B with index pass through. |
| 13 | Output - Step/Dir with index pass through. |
| 14 | Output - CW/CCW with index pass through. |
| 15 | Output - A/B stepper control. |
| 16 | Output - Step/Dir stepper control. |
| 17 | Output - CW/CCW stepper control. |

NOTE

If you are using multi-turn or single turn absolute feedback devices, and the once per rev or absolute Z pulse modes (DRV.EMUEMODES 1,2,6,7,8, or 9) the Z pulse generated by the EEO will always be aligned with the same mechanical position of the primary feedback position. If you are using an incremental feedback device and these modes, then the origin of the primary feedback is not at the same mechanical position each time the drive powers up. If you are using the pass through mode (DRV.EMUEMODES 12, 13, or 14) the Z pulse generated by the EEO will always be aligned with the same mechanical position of the primary feedback position.

Related Topics

DRV.EMUERES (→ p. 396), DRV.EMUEZOFFSET (→ p. 400), DRV.EMUEMTURN (→ p. 394)

19.10.25 DRV.EMUEMTURN

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Defines the location of the index pulse on the EEO (emulated encoder output) when DRV.EMUEMODE=2. |
| Units | revolutions |
| Range | 0 to 4,294,967,295 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3491h/0 | M_01-00-00-000 |

Description

When the emulated encoder output (EEO) is configured to generate an absolute index pulse (DRV.EMUEMODE is 2, 7 or 9) this parameter and DRV.EMUEZOFFSET define the location of the Z pulse. DRV.EMUEMTURN is used to define which turn of the position range the Z pulse is located. DRV.EMUEZOFFSET is used to define the position of the Z pulse within one revolution.

Note: If you are using multi-turn or single tune absolute feedback devices the Z pulse from generated by the EEO will always be aligned with the same mechanical position of the of the primary feedback position. If you are using an incremental feedback device then the origin of the primary feedback is not at the same mechanical position each time the drive powers up.

Related Topics

DRV.EMUEMODE (→ p. 392), DRV.EMUERES (→ p. 396)

19.10.26 DRV.EMUEPULSEWIDTH

| General Information | |
|---------------------|---|
| Type | TBD |
| Description | Sets the encoder output pulse width for modes 6 to 7. |
| Units | us(microseconds) |
| Range | 4.08 us to 2621.48 us |
| Default Value | 4.08 us |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 520Dh/0 | M_01-00-00-000 |

Description

EMU Encoder Out Pulse Width Register

Sets pulse width for CW/CCW and Step and Direction modes. This parameter does not effect the A quad B mode. To calculate emuOutPulseWidth:

(Desired pulse width - 40 nsec)/520 nsec

| Requirement | | DSFPGA-03-306 |
|-------------|------------------|---|
| Bits | Bits | Description |
| 11:0 | emuOutPulseWidth | Read/Write 12 bit unsigned number minimum resolution is 520 nsec. Reset state – 0 |
| 15:12 | | reserved |

- Register is in counts (12 bit)
- Register * 520ns + 40ns is the actual pulse with.
- Register = 1 = pulse width is 560ns = 0.56us (minimum value)
- For each register increment the pulse width is raised by 0.52us

Example

50 usecs pulse width

$$\text{emuOutPulseWidth} = (50 \text{ usecs} - 40 \text{ nsec}) / 520 \text{ nsec} = 96$$

$$\text{actual pulse} = 96 * 520 \text{ nsec} + 40 \text{ nsec} = 49.88 \text{ usecs.}$$

Related Topics

DRV.EMUEMODE (→ p. 392)

19.10.27 DRV.EMUERES

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the resolution of the EEO (emulated encoder output). |
| Units | lines/rev (when DRV.EMUEMODE (→ p. 392) = 1, 2, or 3) counts/rev (when DRV.EMUEMODE (→ p. 392) = 4 or 5) |
| Range | 0 to 16,777,215 lines per revolution |
| Default Value | 0 lines per revolution |
| Data Type | Integer |
| Start Version | M_01-00-00-000 (resolution increased from 65,535 to 16,777,215 in M_01-04-00-000) |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3535h/0 | M_01-00-00-000 |

Description

This parameter sets the emulated encoder (EEO) resolution. DRV.EMUERES also defines how many lines are output for one revolution of the primary feedback (when this port is configured as an output), or how many lines will be considered a full revolution of the handwheel (when this port is configured as an input).

Related Topics

DRV.EMUEMODE (→ p. 392)

19.10.28 DRV.EMUESTEPCMD

| General Information | |
|---------------------|--|
| Type | R/W |
| Description | Setpoint for stepper motor output through the emulated encoder output. |
| Units | None |
| Range | -2147483648 to 2147483648 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-14-02-000 |

Description

If using emulated encoder output modes 15, 16 or 17 (stepper control), the output signals on the X9 connector can be controlled through EtherCAT/CANopen objects. It is recommended to update the setpoint values using cyclic data (PDO) to create a smooth trajectory for the stepper command.

Related Topics

Stepper Control and EEOs over fieldbus

DRV.EMUESTEPCMDPIN (→ p. 398) | DRV.EMUESTEPCMDPOUT (→ p. 399)

19.10.29 DRV.EMUESTEPCMDPIN

| General Information | |
|---------------------|---|
| Type | R/W |
| Description | Scaling factor (numerator) for the command DRV.EMUESTEPCMD. |
| Units | N/A |
| Range | -65536 to 65536 |
| Default Value | 65536 |
| Data Type | Integer |
| Start Version | M_01-14-03-000 |

Description

This parameter is the numerator in the scaling factor for DRV.EMUSTEPCMD.

The internal stepper command is calculated as follows:

Related Topics

Stepper Control and EEOs over fieldbus

DRV.EMUESTEPCMD (→ p. 397) | DRV.EMUESTEPCMDPOUT (→ p. 399)

$$\frac{DRV.EMUESTEPPIN}{DRV.EMUESTEPPOUT} * DRVEMUESTEPCMD$$

19.10.30 DRV.EMUESTEPCMDPOUT

| General Information | |
|---------------------|---|
| Type | R/W |
| Description | Scaling factor (denominator) for the command DRV.EMUESTEPCMD. |
| Units | N/A |
| Range | -65536 to 65536 |
| Default Value | 65536 |
| Data Type | Integer |
| Start Version | M_01-14-03-000 |

Description

This parameter is the denominator in the scaling factor for DRV.EMUSTEPCMD.

The internal stepper command is calculated as follows:

Related Topics

Stepper Control and EEOs over fieldbus

DRV.EMUESTEPCMD (→ p. 397) | DRV.EMUESTEPCMDPIN (→ p. 398)

$$\frac{DRV.EMUESTEPPIN}{DRV.EMUESTEPPOUT} * DRVEMUESTEPCMD$$

19.10.31 DRV.EMUZOFFSET

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the location of the EEO (emulated encoder output) index pulse (when DRV.EMUEMODE=1). |
| Units | 1/65536 rev |
| Range | 0 to 65535 rev |
| Default Value | 0 rev |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3537h/0 | M_01-00-00-000 |

Description

When emulated encoder output (EEO) multiturn is selected (DRV.EMUEMODE (→ p. 392)=1), this parameter is used by itself to define the position of the Z pulse within one revolution. When the primary feedback position (within a revolution) equals this value, an index pulse will output. Also, if DRV.EMUEMODE=1 then this parameter is used in conjunction with DRV.EMUEMTURN.

When the EEO is configured to generate an absolute index pulse (DRV.EMUEMODE is 2, 7 or 9) this parameter and DRV.EMUZOFFSET define the location of the Z pulse. DRV.EMUEMTURN is used to define which turn of the position range the Z pulse is located and DRV.EMUZOFFSET is used to define the position of the Z pulse within one revolution.

Note: If you are using multi-turn or single turn absolute feedback devices the Z pulse from generated by the EEO will always be aligned with the same mechanical position of the of the primary feedback position. If you are using an incremental feedback device then the origin of the primary feedback is not at the same mechanical position each time the drive powers up.

Related Topics

DRV.EMUEMODE (→ p. 392), DRV.EMUEMTURN (→ p. 394)

19.10.32 DRV.EN

| General Information | |
|---------------------|--|
| Type | Command |
| Description | Enables the axis (software). |
| Units | N/A |
| Range | N/A |
| Default Value | Analog drive software is enabled. All other types of drive software are disabled. |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

DRV.EN issues a software enable to the drive. You can query the value of DRV.ACTIVE (→ p. 368) to check whether the drive is currently enabled or disabled.

You can also query the value of DRV.DISSOURCES (→ p. 386) to check whether the software enable bit is high (software enabled was issued by executing DRV.EN) or the software enable bit is low (software disable was issued by executing DRV.DIS). If the drive software enable bit is low and DRV.EN is executed, then drive faults are automatically cleared during the software enable process.

Related Topics

15.4 Clearing Faults

10.3 Enable/Disable

DRV.DIS (→ p. 384), DRV.DISSOURCES (→ p. 386), DRV.ACTIVE (→ p. 368)

19.10.33 DRV.ENDEFAULT

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the default state of the software enable. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Boolean |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5080h/0 | M_01-00-00-000 |

Description

DRV. ENDEFAULT sets the default state of the software enable on power up for drives without fieldbuses (DRV.CMDSOURCE other than 1).

NOTE

It is recommended to not use this parameter with a BASIC program (keep DRV.ENDEFAULT = 0 / default value). Instead set DRV.SWENABLE = 1 at the beginning of the BASIC program.

Related Topics

Enable/Disable (→ p. 77)

19.10.34 DRV.ERRORLIST

| General Information | |
|---------------------|---|
| Type | R/O |
| Description | Prints a list of all possible error codes and error text. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-09-00-000 |

Description

AKD PDMM commands and parameters reports an error code when a command cannot be executed. Use DRV.ERRORLIST to print a list of all possible error codes and their descriptions.

19.10.35 DRV.FAULTDISPLAYMODE

| General Information | |
|---------------------|------------------------------|
| Type | NV Parameter |
| Description | Sets the fault display mode. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-13-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 53C7h | M_01-13-00-000 |

Description

DRV.FAULTDISPLAYMODE sets the LED display behavior when faults are present.

| Value | Description |
|-------|---|
| 0 | Only the highest priority fault is displayed. |
| 1 | All faults displayed. |

Related Topics

DRV.CLRFAULTS (→ p. 373), DRV.FAULTHIST (→ p. 405), DRV.CLRFAULTHIST (→ p. 372)

19.10.36 DRV.FAULTHIST

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the last 50 faults from NV memory. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 2012h | M_01-00-00-000 |
| | 2013h | |

Description

DRV.FAULTHIST returns the last 50 faults that occurred in the drive. The faults are shown with their fault number (which matches the one displayed on the drive display) and a time stamp that indicates when they last occurred.

Issue a DRV.CLRFAULTHIST (→ p. 372) to clear this fault log.

Related Topics

DRV.FAULTS (→ p. 407), DRV.CLRFAULTHIST (→ p. 372)

19.10.37 DRV.FAULT1 to DRV.FAULT10

| General Information | |
|---------------------|--|
| Type | R/O |
| Description | Location of fault codes for any active fault conditions. |
| Units | N/A |
| Range | Any supported fault code or 0. |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-06-00-000 |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|-------------|----------------------|
| EtherCAT COE and CANopen | 2001h/1 | DRV.FAULT1 | M_01-00-00-000 |
| | 2001h/2 | DRV.FAULT2 | |
| | 2001h/3 | DRV.FAULT3 | |
| | 2001h/4 | DRV.FAULT4 | |
| | 2001h/5 | DRV.FAULT5 | |
| | 2001h/6 | DRV.FAULT6 | |
| | 2001h/7 | DRV.FAULT7 | |
| | 2001h/8 | DRV.FAULT8 | |
| | 2001h/9 | DRV.FAULT9 | |
| | 2001h/A | DRV.FAULT10 | |

Description

These parameters are holding registers where any active faults will be kept. A value of zero represents that no fault is present. Non-zero values correspond to specific fault codes in the drive (see fault and warning messages). The registers are populated in the order of when the fault occurs (DRV.FAULT1, DRV.FAULT2, DRV.FAULT3, and so on).

Notes:

- If DRV.FAULT1 value is 0, then the drive does not have any faults.
- Only active faults are shown. This is not a fault history.
- These registers are an alternative to the string type parameter DRV.FAULTS (→ p. 407), so that field-busses and AKD BASIC users have easier access to the details of the faults in the drive.
- Warnings are not shown in the registers, only faults.

Related Topics

DRV.ACTIVE (→ p. 368) | DRV.WARNING1 to DRV.WARNING3 (→ p. 443)

19.10.38 DRV.FAULTS

| General Information | |
|---------------------|--------------------------|
| Type | R/O Parameter |
| Description | Reads the active faults. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

Description

DRV.FAULTS returns a list of all currently active faults in the system, preceded by their fault number which matches the number displayed on the drive display.

To clear the faults, either issue a DRV.CLRFAULTS or issue a DRV.DIS followed by DRV.EN.

If no active faults are in the system, then after executing DRV.CLRFAULTS the value read by DRV.FAULTS is "No faults active".

Example

```
-->DRV.FAULTS
502: Bus under voltage.
-->
```

Related Topics

DRV.CLRFAULTS (→ p. 373), DRV.FAULTHIST (→ p. 405), DRV.CLRFAULTHIST (→ p. 372)

19.10.39 DRV.HANDWHEELSRC

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Selects the feedback for handwheel operation. |
| Units | None |
| Range | 2-3 |
| Default Value | 2 |
| Data Type | U8 |
| Start Version | M_01-05-08-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5263h/0 | M_01-05-08-000 |

Description

This command sets the feedback which will be used as the handwheel source. If the selected Feedback is incompatible with the selected emulated encoder mode, a warning will be displayed.

Feedback 3 is only supported on drives with model numbers similar to AKD PDMM-x-xxxxx-NBxx-xxxx and will only work with Endat 2.2 multiturm encoder.

19.10.40 DRV.HELP

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the minimum, maximum, and default values for a specific parameter or command. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

Description

This parameter returns more information about a specific parameter or command.

In most cases, except special parameters, this command tells you the minimum, maximum, default, and actual value of a parameter. Exceptions are commands that do not have these values (such as DRV.EN (→ p. 401)) or information commands (such as DRV.VER (→ p. 441)).

Related Topics

Terminal (→ p. 169)

19.10.41 DRV.HELPALL

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Retrieves the minimum, maximum, default, and actual values for all available parameters and commands. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

Description

This parameter retrieves all information about all parameters and commands in the firmware. In most cases, DRV.HELPALL returns the minimum, maximum, default, and actual value for each parameter and command. Exceptions include parameters and commands that do not have these values (such as DRV.EN (→ p. 401)) or pure INFO commands (such as DRV.VER (→ p. 441)).

Related Topics

Terminal (→ p. 169)

19.10.42 DRV.HWENABLE

| General Information | |
|---------------------|--------------------------------|
| Type | R/O |
| Description | Status of the hardware enable. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | tbd |

Description

Status of the Hardware Enable.

0 - not enabled

1 - enabled

Notes: This parameter reflects the status of the hardware enable only, not if the status of the power stage. The status of the power stage enable is determined by DRV.ACTIVE.

Related Topics

DRV.DISSOURCES (→ p. 386) | DRV.ACTIVE

19.10.43 DRV.HWENDELAY

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Delay time between inactive Hardware Enable input and drive disable. |
| Units | Milliseconds |
| Range | 0 to 167 ms |
| Default Value | 0 ms |
| Data Type | Integer |
| Start Version | 01-05-08-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5264h/0 | M_01-05-08-000 |

Description

By default, when the Hardware Disable input deactivates, the drive is immediately disabled. However, on a vertical axis this may allow the load to drop slightly before the brake is applied.

To ensure that the brake is applied before the power stage is disabled, set DRV.HWENDELAY to a value which allows the brake to apply fully.

While the brake is applying, the drive will also attempt to decelerate the motor using the standard drive disable settings such as DRV.DISMODE, CS.DEC, CS.VTHRESH and CS.TO.

NOTE

Prior to version 01-05-08-000, the brake would only apply when velocity dropped below CS.VTHRESH or MOTOR.TBRAKETO expired. Starting with 01-05-08-000, the brake will now apply immediately when the Hardware Enable Input line is deactivated.

Related Topics

DRV.DISMODE (→ p. 385) | CS.DEC (→ p. 329) | CS.VTHRESH (→ p. 332) | CS.TO (→ p. 331) | MOTOR.TBRAKEAPP (→ p. 622)

19.10.44 DRV.HWENMODE

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Selects the action that the hardware enable digital input will perform. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-03-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3506h/0 | M_01-03-00-000 |

Description

This parameter selects the action that the hardware enable digital input will perform.

0 = The rising edge of the hardware enable will clear the drive faults.

1 = The rising edge of the hardware enable will not clear the drive faults.

The high/low state of the hardware enable is always used to control the active enable state of the drive.

Related Topics

10.3 Enable/Disable

19.10.45 DRV.ICONT

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the continuous rated current value. |
| Units | Arms |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5083h/0 | M_01-00-00-000 |

Description

DRV.ICONT returns the drive continuous rated current in Arms.

Related Topics

DRV.IPEAK (→ p. 417)

19.10.46 DRV.INFO

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads general information about the drive. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-00-00-000 |

Description

DRV.INFO returns general information about the drive.

Example

```
Advanced Kollmorgen Drive
```

```
-----
```

```
Drive model : AKD PDMM-P00306-NACC-0000
```

```
Drive type : Position Indexer
```

```
Continuous current : 3.000 Arms
```

```
Peak current: 9.000 Arms
```

```
Voltage : 120/240 VAC
```

```
Option Board : Not applicable
```

```
Connectivity : EtherCAT
```

```
Product serial number: R-0939-00048
```

```
Hardware version : --
```

```
Mac address: 00-23-1B-00-50-F1
```

```
Processor ID: 0xE5040003
```

```
Firmware version: M_01-04-16-000_z_2011-09-12_14-03-45_AP
```

```
Operational image : M_01-04-16-000_z_2011-09-12_14-03-45_AP
```

```
Resident image : R_00-00-28-000
```

```
Revision : 19074
```

```
Source Location: local
```

```
FPGA Version: FP0003_0103_00_00
```

```
Operational image : FP0003_0103_00_00
```

Resident image : FPB003_0100_00_00

Size: 1600

Control board serial number : 4-0921-00196

Part number : 0

Revision : 7

Board ID : Standard

Power board serial number : 4-0922-00156

Part number : 0

TCP/IP IP address : 169.254.250.241

Subnet mask : 255.255.0.0

Default gateway : 0.0.0.0

DHCP server : 0.0.0.0

Temporary fieldbus type : EtherCAT

FPGA size : 1600

19.10.47 DRV.IPEAK

| General Information | |
|---------------------|-------------------------------------|
| Type | R/O Parameter |
| Description | Reads the peak rated current value. |
| Units | Arms |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5084h/0 | M_01-00-00-000 |

Description

DRV.IPEAK returns the drive peak rated current in Arms.

Related Topics

7.5 Foldback

DRV.ICONT (→ p. 414)

19.10.48 DRV.IZERO

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the current that will be used during the DRV.ZERO procedure. |
| Units | Arms |
| Range | Drive peak current to 0 Arms |
| Default Value | 0 Arms |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5085h/0 | M_01-00-00-000 |

Description

This parameter sets the current that is used during the DRV.ZERO (→ p. 445) procedure.

Related Topics

DRV.ZERO (→ p. 445)

19.10.49 DRV.LIST

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the list of available parameters and commands. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-00-00-000 |

Description

DRV.LIST reads the list of available commands and parameters from the drive.

To filter this list, enter DRV.LIST followed by the prefix of the commands and parameters that you wish to display.

Example

Return a list of all available commands in the system:

```
-->DRV.LIST
```

Return all commands with the prefix DRV:

```
-->DRV.LIST DRV
```

19.10.50 DRV.LOGICVOLTS

| General Information | |
|---------------------|---------------------------|
| Type | R/O Parameter |
| Description | Reads the logic voltages. |
| Units | mv , Ω |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-00-00-000 |

Description

DRV.LOGICVOLTS reads the logic voltages data of 1.2 V, 2.5 V, 3.3 V, 5 V, 12 V, -12 V, and 3.3 AV.

Example

Below is an example of the output for this command:

```

ch0 = 1.2V      : 1211 mv
ch1 = 2.5V      :2488 mv
ch2 = 3.3V      :3274 mv
ch3 = 5V        :4950 mv
ch4 = 12V       :11892 mv
ch5 = -12V      : -11912 mv
ch6 = 3.3AV     :3300 mv
ch7 = R ohm     :100000 ohm

```

19.10.51 DRV.MEMADDR

| General Information | |
|---------------------|----------------------------------|
| Type | R/W Parameter |
| Description | Sets the read and write address. |
| Units | N/A |
| Range | N/A |
| Default Value | U8 |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

Description

DRV.MEMADDR sets the address that is used by DRV.MEMDATA. The input can be either an internal parameter of the drive or any direct address from the DSP address space (SDRAM, internal RAM, or asynchronous memory). The input value can be either decimal or hexadecimal with 0x prefix.

Type extension can be one of the following:

U8,S8,U16,S16,U32,S32,U64,S64.

Examples

Setting to an internal parameter:

```
-->DRV.MEMADDR CCommandHandler.Debug1
```

Setting to an internal address:

```
-->DRV.MEMADDR 0xffabcde.u16
```

Related Topics

DRV.MEMDATA (→ p. 422)

19.10.52 DRV.MEMDATA

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets or reads a value from an internal address. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

Description

DRV.MEMDATA reads or writes a value using the address set by DRV.MEMADDR (→ p. 421). The write value accepts decimal or "0x" hexadecimal format. The read value defaults to decimal display but can be set to hexadecimal with the command "DRV.READFORMAT (→ p. 431) 16".

Examples

WARNING

These are example addresses. Never write to an address unless you are certain it is safe.

Read a value from internal address:

```
-->DRV.MEMADDR 0x1234.u32
--DRV.MEMDATA
3800219712
```

Write a decimal value to an internal address:

```
-->DRV.MEMDATA 123
```

Write a hexadecimal value to an internal address:

```
-->DRV.MEMDATA 0xFEDC
```

19.10.53 DRV.MOTIONDISSOURCES

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Returns some causes of a motion disable. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-10-03-000 |

Description

This parameter reports some of the reasons that motion may not be allowed on the axis. There are additional reasons why the drive cannot accept a motion command that are not reported by this command (for example, fieldbus control bits or fault status).

| Value | Description |
|-------|--|
| 0 | Motion not blocked by reason indicated for value 1, 2, or 3. |
| 1 | Suspend Motion Active |
| 2 | Brake not released |
| 4 | Brake not released |
| 8 | Rotor flux not ready |

Related Topics

Digital Inputs and Outputs (pg 1)

19.10.54 DRV.NAME

| General Information | |
|---------------------|---------------------------------------|
| Type | NV Parameter |
| Description | Sets and reads the name of the drive. |
| Units | N/A |
| Range | N/A |
| Default Value | No-Name |
| Data Type | String |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 2031h/0 | M_01-00-00-000 |

Description

You can assign a unique name to any drive under the following conditions:

- Only use ASCII characters
- Max length of 10 characters
- No spaces in name

This name is one way to identify the drive in a multiple drive network (for instance, in a TCP/IP network on which multiple drives reside).

From the terminal screen, DRV.NAME returns the name of the drive as ASCII characters.

19.10.55 DRV.NVCHECK

| General Information | |
|---------------------|-----------------------|
| Type | R/O Parameter |
| Description | NV Parameter Checksum |
| Units | None |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-04-12-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 2019h/0 | M_01-04-12-000 |

Description

DRV.NVCHECK returns a checksum of all the drives NV parameters. This parameter can be used to detect changes in parameters.

In some applications a master device needs to confirm the AKD PDMM drive contains an expected set of drive parameters. Reading and checking all the drive parameters individually is viable but this would be a long process involving many reads over the network. DRV.NVCHECK is a checksum on all the NV parameters and this parameter can be read in a single transaction. DRV.NVCHECK will return the same number if all the drive parameters match. If any of the drive parameters are changed then DRV.NVCHECK will return a different value.

Related Topics

DRV.NVLIST (→ p. 426)

19.10.56 DRV.NVLIST

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Lists the NV parameters and values from the RAM. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

Description

DRV.NVLIST lists all the drive parameters that reside in NV memory.

The list includes each parameter name, followed by its current value from the RAM.

19.10.57 DRV.NVLOAD

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Loads all data from the NV memory of the drive into the RAM parameters. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

DRV.NVLOAD loads all data from the NV memory of the drive into the RAM parameters.

Related Topics

DRV.NVLOAD , DRV.NVLIST (→ p. 426)

19.10.58 DRV.NVSAVE

| General Information | |
|---------------------|---|
| Type | Command |
| Description | Saves the drive parameters from the RAM to the NV memory. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 35EBh/0 | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

DRV.NVSAVE saves the current drive parameter values from the RAM to the NV memory.

The drive parameters that were saved to the NV are read from the NV on the next drive boot, causing the values to be automatically set to the saved values on every drive boot.

Executing DRV.RSTVAR does not modify the values of the NV, but instead sets the drive values in RAM to their defaults.

Related Topics

DRV.RSTVAR (→ p. 433)

19.10.59 DRV.ONTIME

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Returns how long the drive has been running since last power up. |
| Units | Days:Hours:Minutes:Seconds |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-00-00-000 |

Description

This parameter returns the length of time that the drive has been running for the current session (since the last power up).

Related Topics

Returns how long the drive has been running since first activated. (→ p. 434)

19.10.60 DRV.OPMODE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the drive operation mode (current, velocity, or position). |
| Units | N/A |
| Range | 0 to 2 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 35B4h/0 | M_01-00-00-000 |

Description

DRV.OPMODE specifies the operation mode of the drive. You must also use DRV.CMDSOURCE to set the source of the command to the drive.

The operation mode values can be set as follows:

| Mode | Description |
|------|---------------------------------|
| 0 | Current (torque) operation mode |
| 1 | Velocity operation mode |
| 2 | Position operation mode |

DRV.OPMODE can be changed while the drive is enabled or disabled. If you are using the terminal to change the operation mode, then it is recommended that you disable the drive before changing the operation mode. If you change the operation mode from the terminal while the drive is enabled, the system may experience a step change in demand.

Example

Set the source of the command to a TCP/IP channel and the desired operation mode to velocity:

```
-->DRV.CMDSOURCE 0
-->DRV.OPMODE 1
```

Related Topics

10.4 Controlled Stop

1 Current Loop

1 Velocity Loop

1 Position Loop

DRV.CMDSOURCE (→ p. 375)

19.10.61 DRV.READFORMAT

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the value returned to either decimal or hexadecimal. |
| Units | N/A |
| Range | 10 or 16 |
| Default Value | 10 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

Description

DRV.READFORMAT sets the return values type to either decimal or hexadecimal.

| Format | Description |
|--------|--|
| 10 | Sets the read values to decimal format |
| 16 | Sets the read values to hexadecimal format |

19.10.62 DRV.REBOOT

| General Information | |
|---------------------|-----------------------------|
| Type | Command |
| Description | Performs a software reboot. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-13-08-000 |

Description

This parameter performs a software reboot when the following command is issued in the EWV¹ terminal:

```
DRV.REBOOT 0xAABBCCDD
```

EWV will disconnect from the AKD PDMM when the reboot command is executed. The AKD PDMM can be reconnected to EWV approximately one minute after the reboot command is issued. The parameters stored in non-volatile memory will be restored to the AKD PDMM once rebooted.

¹Embedded Workbench Views

19.10.63 DRV.RSTVAR

| General Information | |
|---------------------|--|
| Type | Command |
| Description | Sets default values in the drive without re-booting the drive and without resetting the NV memory. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|--------------------|----------------------|
| EtherCAT COE and CANopen | 1011h (→ p. 975)/1 | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

DRV.RSTVAR causes the drive to return to the default values without the need to re-boot the drive first and without resetting the NV memory. Use DRV.RSTVAR to return to the default settings and recover a working drive.

19.10.64 DRV.RUNTIME

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Returns how long the drive has been running since first activated. |
| Units | Days:Hours:Minutes:Seconds |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 2011h | M_01-00-00-000 |

Description

DRV.RUNTIME returns the length of time that the drive has been running since it was first activated. This time includes the current session and the total amount of time from all previous sessions.

19.10.65 DRV.SETUPREQBITS

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the bitwise set status of parameters that must be set before the drive can be enabled. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

Description

This parameter returns the bitwise set status of parameters that needs to be set up before the drive can be enabled. The drive can only be enabled when this parameter returns 0.

NOTE

If MOTOR.AUTOSSET (→ p. 592) is set to 1 (parameters automatically calculated from motor ID data), then all values in the list will be initialized from the feedback device. Otherwise, the parameters must be set manually.

The following table lists the parameters which require setup for each motor type.

| Parameter | Bit | Default Value | Linear Permanent Magnet | Induction Motor V/f Open Loop | Induction Motor Closed Loop | Interior Permanent Magnet | Brush Motor |
|-----------------|-----|----------------------|-------------------------|-------------------------------|-----------------------------|---------------------------|-------------|
| IL.KP | 0 | 50 [V/A] | √ | - | √ | √ | √ |
| MOTOR.IPEAK | 1 | 2 Arms | √ | √ | √ | √ | √ |
| MOTOR.ICONT | 2 | 1 Arms | √ | √ | √ | √ | √ |
| MOTOR.POLES | 4 | 6 | √ | √ | √ | √ | - |
| MOTOR.LQLL | 7 | 17mH | - | - | - | √ | - |
| MOTOR.LDLL | 8 | 17mH | - | - | - | √ | - |
| MOTOR.KE | 9 | 0 | - | - | - | √ | - |
| MOTOR.IMTR | 10 | 100ms | - | - | √ | - | - |
| MOTOR.IMID | 11 | 0 | - | - | √ | - | - |
| MOTOR.VOLTRATED | 12 | 230 Vrms | - | √ | - | - | - |
| MOTOR.VRATED | 13 | 0 | - | - | - | √ | - |
| MOTOR.LISAT | 14 | 25 x DRIVE.IPEAK | - | - | - | √ | - |
| MOTOR.VRIDK1 | 15 | 0 | - | - | - | √ | - |
| MOTOR.VRIDK2 | 16 | 0 | - | - | - | √ | - |
| MOTOR.IDMAX | 17 | 0.5 x MOTOR.IPEAK | - | - | - | √ | - |
| IL.KDPRATIO | 18 | 1 | - | - | - | √ | - |

| Parameter | Bit | Default Value | Linear Per- manent Magnet | Induction Motor V/f Open Loop | Induction Motor Closed Loop | Interior Permanent Magnet | Brush Motor |
|---------------|-----|---------------|---------------------------------|-------------------------------------|--------------------------------------|---------------------------------|----------------|
| MOTOR.PITCH | 19 | 32 | √ | - | - | - | - |
| MOTOR.VOLTMAX | 20 | 480 Vrms | - | - | - | √ | √ |

Related Topics

DRV.SETUPREQLIST (→ p. 437), MOTOR.AUTOSSET (→ p. 592)

19.10.66 DRV.SETUPREQLIST

| General Information | |
|---------------------|--|
| Type | R/O Parameters |
| Description | Reads the list of parameters that must be set before the drive can be enabled. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

Description

This parameter returns the list of parameters that must be set before the drive can be enabled and also whether or not each one of those parameters is set. This list of parameters will vary depending on the value MOTOR.TYPE. The drive can only be enabled when all commands have a value of 0.

NOTE

If MOTOR.AUTOSSET (→ p. 592) is set to 1 (parameters automatically calculated from motor ID data), then all values in the list will be initialized from the feedback device. Otherwise, the parameters must be set manually.

Example

```
-->DRV.SETUPREQLIST
IL.KP 0
MOTOR.ICONT 0
MOTOR.IPEAK 0
MOTOR.POLES 0
-->
```

Related Topics

DRV.SETUPREQBITS (→ p. 435), MOTOR.AUTOSSET (→ p. 592)

19.10.67 DRV.STOP

| General Information | |
|---------------------|--------------------------------------|
| Type | Command |
| Description | This command stops all drive motion. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 35FEh/0 | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

This command stops drive motion. In torque and velocity modes, the command value will immediately be set to 0. In position mode, the axis will decelerate to a full stop. If the position command is from a gearing master, the drive will cancel gearing and decelerate to a stop. This command has no effect if the command source is an analog input (Analog Position, Analog Velocity, or Analog Torque).

If the command values are coming from a fieldbus master, the stop may be immediately overwritten by a new fieldbus command. Use a controlled stop input or disable command to ensure a stop.

19.10.68 DRV.TEMPERATURES

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the temperature of drive components. |
| Units | °C |
| Range | 55 to 125 °C |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3610h/0 | M_01-00-00-000 |
| | 3611h/0 | |

Description

DRV.TEMPERATURES reads the temperature in different parts of the drive (power and control boards). The temperature is read from temperature sensors located in the drive.

Example

Below is an example of the output for this command :

```
Control Temperature: 39 °C
Power1  Temperature: 31 °C
Power2  Temperature: Sensor does not exist.
Power3  Temperature: Sensor does not exist.
```

19.10.69 DRV.TYPE

| General Information | |
|---------------------|---|
| Type | R/W on the CC drive model. R/O on all other drive models. |
| Description | Selects the operational fieldbus on CC drive models. |
| Units | N/A |
| Range | 0 to 8 |
| Default Value | 2 |
| Data Type | Integer |
| Start Version | M_01-03-00-000 |

Description

DRV.TYPE allows you to select the operational fieldbus for your drive. This parameter is read-write on the CC drive models and read-only on all other drive models. To change the operational fieldbus of your drive:

1. Set DRV.TYPE to one of the following values. Only values 0 to 3 are selectable. All other values are read only.
 - 0 = Analog (no EtherCAT or CANopen) with no position indexer functionality.
 - 1 = Analog (no EtherCAT or CANopen) with position indexer functionality.
 - 2 = EtherCAT
 - 3 = CANopen
 - 4 = SynqNet
 - 5 = EtherNet/IP
 - 6 = BASIC Language (not fieldbus)
 - 7 = Profinet
 - 8 = sercos® III
2. Save the parameters to the NV memory on the drive by issuing the DRV.NVSAVE command
3. Power cycle the 24 V supply to the drive. When the drive has finished powering up, it will be working with the new selection.

Changing DRV.TYPE does not immediately change the type of the drive fieldbus selection. You must power cycle the drive to start the drive with the selected functionality.

You cannot use EtherCAT and CANopen at the same time. Use FBUS.TYPE or DRV.INFO to identify the fieldbus currently in use.

DRV.TYPE does not change if you use DRV.RSTVAR.

Related Topics

AKD PDMM Models (→ p. 26)

FBUS.TYPE (→ p. 529), DRV.INFO (→ p. 415)

19.10.70 DRV.VER

| General Information | |
|---------------------|--------------------------|
| Type | R/O Parameter |
| Description | Reads the drive version. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-00-00-000 |

Description

DRV.VER reads both FPGA and firmware versions.

The version data presented is hard coded in the firmware code.

Example

Below is an example of the output for this command:

```
Danaher Motion - Digital Servo Drive
-----
FPGA version : FP0004_0001_00_07
Firmware Version : M_0-0-15_T_2009-01-19_10-36-28_IR
```

19.10.71 DRV.VERIMAGE

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Returns the version data from each image. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-00-00-000 |

Description

DRV.VERIMAGE reads the versions of the different images in the drive. This parameter returns the version data from each image .i00 file.

Example

Below is an example of the output for this parameter:

```

Danaher Motion - Digital Servo Drive
-----
Resident Firmware: R_0-0-11
Operational Firmware: M_0-0-15
Resident FPGA: FPB004_0001_00_07
Operational FPGA : FP0004_0001_00_07

```

19.10.72 DRV.WARNING1 to DRV.WARNING3

| General Information | |
|---------------------|--|
| Type | R/O |
| Description | Location of fault codes for any active warning conditions. |
| Units | N/A |
| Range | Any supported fault code or 0 |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|--------------|----------------------|
| EtherCAT COE and CANopen | 2000h/1 | DRV.WARNING1 | M_01-00-00-000 |
| | 2000h/2 | DRV.WARNING2 | |
| | 2000h/3 | DRV.WARNING3 | |

Description

These parameters are holding registers where any active warnings will be displayed. A value of zero represents that no warning is present. Non-zero values correspond to specific warning codes in the drive (see fault and warning messages). The registers are populated in the order of when the warning occurs (DRV.WARNING1, DRV.WARNING2, DRV.WARNING3, and so on).

Notes:

- If DRV.WARNING1 value is 0, then the drive does not have any faults.
- Only active warnings are shown. This is not a warning history.
- These registers are an alternative to the string type parameter DRV.WARNINGS so that field-buses and AKD BASIC user programs have integer-type parameters to access to the details of the warnings in the drive.
- Faults are not shown in the registers, only warnings.

Related Topics

DRV.FAULT1 to DRV.FAULT10 (→ p. 406) |

19.10.73 DRV.WARNINGS

| General Information | |
|---------------------|----------------------------|
| Type | R/O Parameter |
| Description | Reads the active warnings. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| Start Version | M_01-00-00-000 |

Description

DRV.WARNINGS returns a list of all currently active warnings in the system.

19.10.74 DRV.ZERO

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the zero mode. The procedure is activated when the drive is enabled. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

Description

The zero procedure is a sequence in which phase commutation is initialized. During this procedure, the motor is held at a certain known electrical position (by applying a current defined by DRV.IZERO (→ p. 418)). After the motor rests at this position, the commutation angle is calculated and set automatically.

Related Topics

DRV.IZERO (→ p. 418)

19.11 DS402 Parameters

This section describes the DS402 parameters.

| | |
|--|------------|
| 19.11.1 DS402.1ADDPOSFCFEED | 447 |
| 19.11.2 DS402.1ADDPOSFCSHAFTREV | 448 |
| 19.11.3 DS402.3ADDPOSFCFEED | 449 |
| 19.11.4 DS402.3ADDPOSFCSHAFTREV | 450 |

19.11.1 DS402.1ADDPOSFCFEED

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the 1st additional feed constant - feed. |
| Units | N/A |
| Range | 1 to 4294967295 |
| Default Value | 1 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 60E9h/1 | M_01-00-00-000 |

Description

Sets the 1st additional feed constant - feed.

Related Topics

DS402.1ADDPOSFCSHAFTREV (→ p. 448)

19.11.2 DS402.1ADDPOSFCSHAFTREV

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the 1st additional feed constant - driving shaft revolutions. |
| Units | N/A |
| Range | 1 to 4294967295 |
| Default Value | 1 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 60EEh/1 | M_01-00-00-000 |

Description

Sets the 1st additional feed constant - driving shaft revolutions.

Related Topics

DS402.1ADDPOSFCFEED (→ p. 447)

19.11.3 DS402.3ADDPOSFCFEED

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the 3rd additional feed constant - feed. |
| Units | N/A |
| Range | 1 to 4294967295 |
| Default Value | 1 |
| Data Type | Integer |
| Start Version | M_013-11-02-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 60E9h/3 | M_013-11-02-000 |

Description

Sets the 3rd additional feed constant - feed.

Related Topics

DS402.3ADDPOSFCSHAFTREV (→ p. 450)

19.11.4 DS402.3ADDPOSFCSHAFTREV

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the 3rd additional feed constant - driving shaft revolutions. |
| Units | N/A |
| Range | 1 to 4294967295 |
| Default Value | 1 |
| Data Type | Integer |
| Start Version | M_013-11-02-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 60EEh/3 | M_013-11-02-000 |

Description

Sets the 3rd additional feed constant - driving shaft revolutions.

Related Topics

DS402.3ADDPOSFCFEED (→ p. 449)

19.12 FAULT Parameters

This section describes the FAULT parameters.

| | |
|--------------------------------------|------------|
| 19.12.1 FAULTx.ACTION | 452 |
| 19.12.2 FAULT570.ACTION | 453 |

19.12.1 FAULTx.ACTION

| General Information | |
|---------------------|---|
| Type | R/W |
| Description | Gets/Sets the Fault Action for Fault 130, 131, 132, 134, 139, 451, and 702. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-04-16-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5259h/0 | FAULT130.ACTION |
| | 525Ah/0 | FAULT131.ACTION |
| | 525Bh/0 | FAULT132.ACTION |
| | 525Ch/0 | FAULT134.ACTION |
| | 5267h/0 | FAULT451.ACTION |
| | 525Dh/0 | FAULT702.ACTION |
| | | M_01-04-16-000 |

Description

This Parameter determines the action the drive should take when Fault 130, 131, 132, 134, 139, 451, or 702 occurs.

| Parameter Value | Drive Action |
|-----------------|-------------------------------------|
| 0 | Disable Amplifier |
| 1 | Ignore (fault will not be reported) |

19.12.2 FAULT570.ACTION

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Gets/Sets the fault action for fault 570. |
| Units | N/A |
| Range | 0, 1, 4 |
| Default Value | 4 |
| Data Type | Integer |
| Start Version | M_01-013-00-000 |

Variants Supported

Supported on all HV48 drive types.

Description

This parameter sets the action performed when F570 (→ p. 192) occurs.

| Value | Action |
|-------|------------------|
| 0 | Generate Fault |
| 1 | Ignore |
| 4 | Generate warning |

Related Topics

Fault and Warning Messages (→ p. 177)

19.13 FB1 Parameters

This section describes the FB1 parameters.

| | |
|--|------------|
| 19.13.1 FB1.BISSBITS | 456 |
| 19.13.2 FB1.CALTHRESH | 457 |
| 19.13.3 FB1.DIAG | 458 |
| 19.13.4 FB1.ENCREC | 459 |
| 19.13.5 FB1.EXTENDEDMULTITURN | 460 |
| 19.13.6 FB1.FAULTS | 461 |
| 19.13.7 FB1.HALLSTATE | 463 |
| 19.13.8 FB1.HALLSTATEU | 464 |
| 19.13.9 FB1.HALLSTATEV | 465 |
| 19.13.10 FB1.HALLSTATEW | 466 |
| 19.13.11 FB1.IDENTIFIED | 467 |
| 19.13.12 FB1.INITPSAVED | 468 |
| 19.13.13 FB1.INITPSTATUS | 469 |
| 19.13.14 FB1.INITPWINDOW | 470 |
| 19.13.15 FB1.INITSIGNED | 471 |
| 19.13.16 FB1.LASTIDENTIFIED | 472 |
| 19.13.17 FB1.MECHPOS | 473 |
| 19.13.18 FB1.MEMVER | 474 |
| 19.13.19 FB1.MOTORPHASE | 475 |
| 19.13.20 FB1.MOTORPOLES | 476 |
| 19.13.21 FB1.OFFSET | 477 |
| 19.13.22 FB1.ORIGIN | 478 |
| 19.13.23 FB1.P | 480 |
| 19.13.24 FB1.PDIR | 481 |
| 19.13.25 FB1.PFIND | 482 |
| 19.13.26 FB1.PFINDCMDU | 483 |
| 19.13.27 FB1.PIN | 484 |
| 19.13.28 FB1.POFFSET | 485 |
| 19.13.29 FB1.POLES | 486 |
| 19.13.30 FB1.POUT | 487 |
| 19.13.31 FB1.PSCALE | 488 |
| 19.13.32 FB1.PUNIT | 489 |

| | |
|--|------------|
| 19.13.33 FB1.RESKTR | 490 |
| 19.13.34 FB1.RESREFPHASE | 491 |
| 19.13.35 FB1.SELECT | 492 |
| 19.13.36 FB1.TRACKINGCAL | 494 |
| 19.13.37 FB1.USERBYTE0 to FB1.USERBYTE7 | 495 |
| 19.13.38 FB1.USERDWORD0 to FB1.USERDWORD1 | 497 |
| 19.13.39 FB1.USERWORD0 to FB1.USERWORD3 | 499 |

19.13.1 FB1.BISSBITS

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Specifies the number of Biss Sensor (Position) Bits for the BiSS Mode C encoder in use. |
| Units | bits |
| Range | 0 to 64 bits |
| Default Value | 32 bits |
| Data Type | Integer |
| See Also | FB1.SELECT (→ p. 492), FB1.IDENTIFIED (→ p. 467) |
| Start Version | M_01-01-00-100 and M_01-01-03-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 508Ch/0 | M_01-00-00-000 |

Description

FB1.BISSBITS specifies the number of BiSS sensor (position) bits for the BiSS Mode C encoder in use. Typically the value is either 26 or 32 for a BiSS Mode C Renishaw encoder. The required value for this parameter is provided by the feedback device manufacturer for the particular device being used.

Related Topics

Feedback 1 (→ p. 54)

19.13.2 FB1.CALTHRESH

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the velocity threshold at which the auto-calibration function will continuously update error correction for sin/cos encoders. |
| Units | Depends on UNIT.VROTARY (→ p. 1) or UNIT.VLINEAR (→ p. 1) Rotary: rpm, rps, deg/s, (custom units)/s, rad/s Linear: counts/s, mm/s, µm/s, (custom units)/s |
| Range | 0 to maximum velocity |
| Default Value | 117.000 rpm |
| Data Type | Float |
| Start Version | M_01-05-11-000 |

Description

FB1.CALTHRESH sets the velocity threshold at which the auto-calibration function will continuously update error correction for sin/cos encoders.

Related Topics

Feedback 1 (→ p. 54)

19.13.3 FB1.DIAG

| General Information | |
|---------------------|--|
| Type | R/O |
| Description | Returns feedback diagnostic information. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-08-00-006 |

Description

Some fault documentation may reference this keyword to collect additional information. The results of this keyword vary for each situation.

If you have been directed to use this keyword, please collect the results and contact Kollmorgen support for the next steps in resolving your problem.

Related Topics

FB1.FAULTS (→ p. 461)

19.13.4 FB1.ENCRES

| General Information | |
|---------------------|--|
| Type | Depends on FB1.IDENTIFIED. See table in description below. |
| Description | Sets the resolution of the motor encoder. |
| Units | Encoder counts |
| Range | 0 to $2^{32}-1$ |
| Default Value | 1,024 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3533h/0 | M_01-00-00-000 |

Description

This parameter sets or gets the resolution of the motor encoder (encoder feedback systems only) in number of counts per revolution for a rotary motor and the number of encoder pitches per motor pole pitch for a linear motor. The number of encoder counts per revolution is obtained by multiplying the motor catalog resolution in units of PPR by four. For example, for a 1024 PPR resolution motor, the number of encoder counts per revolution is $1024 * 4 = 4096$. For this motor FB1.ENCRES must be set to 4096.

For linear motors, the value of FB1.ENCRES is set to the number of encoder pitches per motor pole pitch. For a motor with 32 mm pole pitch, and a 40 μm encoder pitch, the value for FB1.ENCRES should be set to $32 \text{ mm} / 40 \mu\text{m} = 800$.

Depending on the value of FB1.IDENTIFIED, FB1.ENCRES changes between read-only and read-write. The following table lists the FB1.IDENTIFIED values and FB1.ENCRES's corresponding type.

| FB1.IDENTIFIED value | FB1.ENCRES type |
|------------------------------------|-----------------|
| 10 (Incremental encoder) | R/W |
| 11 (Incremental encoder, no halls) | R/W |
| 20 (Sine encoder) | R/W |
| 21 (Sine encoder, no halls) | R/W |
| 30 (Endat 2.1) | R/O |
| 31 (Endat 2.2) | R/O |
| 32 (biSS) | R/O |
| 33 (hiperface) | R/O |
| 34 (biSS Mode C) | R/W |
| 40 (Resolver) | R/W |
| 41 (sfd) | R/O |
| 42 (Tamagawa) | R/O |

Related Topics

7.2 Feedback 1

19.13.5 FB1.EXTENDEDMULTITURN

| General Information | |
|---------------------|--|
| Type | R/W |
| Description | Enables saving of the extended multiturn position. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-013-08-000 |

Description

This parameter allows additional multiturn information to be stored in non-volatile memory and restored on power up.

Related Topics

Extended Multiturn (→ p. 57)

19.13.6 FB1.FAULTS

| General Information | |
|---------------------|--|
| Type | R/O |
| Description | Returns a string describing the root cause of Fault 467. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-08-00-006 |

Description

Use this parameter to determine the root cause of feedback fault 467. This parameter can currently be used to diagnose faults on Hiperface DSL or SFD3 feedback types. FB1.FAULTS will return one of the following strings as a root cause for fault 467.

| Hiperface DSL Strings |
|--|
| 1000: A log reset was carried out |
| 1001: Acceleration overrun, position invalid |
| 1002: Test in process |
| 1004: Internal error with angle tracking, position invalid |
| 1005: Internal error with vector length, position invalid |
| 1006: Internal error with position counter, position invalid |
| 1007: Internal error with position synchronization, position invalid |
| 1010: Error in absolute position with one rotation |
| 1011: Error 1 in absolute position with multiple rotations |
| 1012: Error 2 in absolute position with multiple rotations |
| 1013: Error 3 in absolute position with multiple rotations |
| 1020: Power on self-test done (only safety-version) |
| 1021: Warning safety-parameter: error has been rectified (only safety-version) |
| 1022: Error safety-parameter: error could not be rectified (only safety-version) |
| 1023: Error calibration-data |
| 1024: Internal communication error 1 |
| 1025: Internal communication error 2 |
| 1026: Internal general error |
| 1030: Critical temperature |
| 1031: Critical LED current |
| 1032: Critical supply voltage |
| 1033: Critical speed |
| 1035: Critical overrun |
| 1036: Internal checking error |

Hiperface DSL Strings

| |
|---|
| 1040: Invalid argument input during resources access |
| 1041: Resources access refused due to incorrect access level |
| 1042: Internal error on resources access |
| 1043: Error on access to user file |
| 1100: No link |
| 1101: No sync |
| 1102: Fault in short command |
| 1103: No FPGA support |
| 1105: Long command read timeout |
| 1106: Long command poll timeout |
| 1107: Spinlock timeout |
| 1110: Long command failure (Refer to FB1.DIAG (→ p. 458) if Fault 1110 is returned) |
| 1111: Resource ID timeout |

SFD3 Strings

| |
|----------------------------|
| 2100: Receive CRC Fault |
| 2102: Communications Fault |

Related Topics

FB1.DIAG (→ p. 458)

19.13.7 FB1.HALLSTATE

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the Hall switch values (encoder feedback only). |
| Units | Terminal: Binary Scope: N/A |
| Range | Terminal: See description below Scope: 1 to 6 |
| Default Value | N/A |
| Data Type | Terminal: String Scope: Integer |
| Start Version | M_01-00-00-000 |

Description

FB1.HALLSTATE reads the Hall switch values (encoder feedback only).

Terminal:

When called from the terminal, FB1.HALLSTATE will return three digits as 0 or 1 (ex. 0 0 1), where the left-most digit is Hall W, the middle digit is Hall V, and the rightmost is Hall U.

0 is a low voltage state, and 1 is a high voltage state on the hall effect.

Scope:

When viewing FB1.HALLSTATE from the scope screen, the result is a single integer. This integer is the sum of the three hall bit states, where $FB1.HALLSTATE = Hall\ U + Hall\ V + Hall\ W$. If the hall is not active, it will return 0. When a hall is active, each hall will contribute the following value to the sum:

- Hall W = 1
- Hall V = 2
- Hall U = 4

| Value | Hall W | Hall V | Hall U |
|-------|--------|--------|--------|
| 1 | √ | - | - |
| 2 | - | √ | - |
| 3 | √ | √ | - |
| 4 | - | - | √ |
| 5 | √ | - | √ |
| 6 | - | √ | √ |

The following sequences indicate the direction of rotation.

| Sequence | Direction |
|---------------|-----------|
| 1,5,4,6,2,3,1 | Positive |
| 1,3,2,6,4,5,1 | Negative |

Related Topics

Feedback 1 (→ p. 54) | FB1.HALLSTATEU (→ p. 464) | FB1.HALLSTATEV (→ p. 465) | FB1.HALLSTATEW (→ p. 466)

19.13.8 FB1.HALLSTATEU

| General Information | |
|---------------------|-----------------------------------|
| Type | R/O Parameter |
| Description | Reads the state of Hall switch U. |
| Units | N/A |
| Range | 0 and 1 |
| Default Value | 1 |
| Data Type | Integer |
| See Also | FB1.HALLSTATE (→ p. 463) |
| Start Version | M_01-03-07-000 |

Description

FB1.HALLSTATEU reads the state of Hall switch U.

Related Topics

Feedback 1 (→ p. 54)

19.13.9 FB1.HALLSTATEV

| General Information | |
|---------------------|-----------------------------------|
| Type | R/O Parameter |
| Description | Reads the state of Hall switch V. |
| Units | N/A |
| Range | 0 and 1 |
| Default Value | 1 |
| Data Type | Integer |
| See Also | FB1.HALLSTATE (→ p. 463) |
| Start Version | M_01-03-07-000 |

Description

FB1.HALLSTATEV reads the state of Hall switch V.

Related Topics

Feedback 1 (→ p. 54)

19.13.10 FB1.HALLSTATEW

| General Information | |
|---------------------|-----------------------------------|
| Type | R/O Parameter |
| Description | Reads the state of Hall switch W. |
| Units | N/A |
| Range | 0 and 1 |
| Default Value | 1 |
| Data Type | Integer |
| See Also | FB1.HALLSTATE (→ p. 463) |
| Start Version | M_01-03-07-000 |

Description

FB1.HALLSTATEW reads the state of Hall switch W.

Related Topics

Feedback 1 (→ p. 54)

19.13.11 FB1.IDENTIFIED

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the type of feedback device used by the drive/motor. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| See Also | FB1.SELECT |
| Start Version | M_01-00-00-000 |

Description

This parameter is set according to FB1.SELECT on drive power up if FB1.SELECT is not -1; otherwise the parameter value is read from the drive memory.

| Type | Description |
|------|---|
| 0 | Unknown |
| 1 | No encoder |
| 10 | Incremental encoder with A/B Quad, marker pulse and Hall |
| 11 | Incremental encoder with A/B Quad, marker pulse and no Hall |
| 12 | Halls Only |
| 20 | Sine Encoder , with marker pulse and Hall |
| 21 | Sine encoder , with marker pulse & No Halls |
| 30 | EnDat 2.1 with Sine Cosine |
| 31 | EnDat 2.2 |
| 32 | BiSS with Sine Cosine |
| 33 | HIPERFACE |
| 34 | BiSS Mode C Renishaw |
| 40 | Resolver |
| 41 | SFD (Smart Feedback Device) |
| 42 | Tamagawa. This feedback type is not available on models with extension type NA (AKD PDMM Models (→ p. 26)). |
| 43 | Network Cyclic Feedback |
| 44 | FB2 Cyclic Feedback |
| 45 | SFD3 |
| 46 | Hiperface DSL |
| 50 | Reserved |
| 63 | Simulated Motor |

Related Topics

Feedback 1 (→ p. 54)

19.13.12 FB1.INITPSAVED

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads position saved at last power down. |
| Units | Depends on UNIT.PLINEAR (pg 1) or UNIT.PROTARY (pg 1) Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, μm , (custom units), 16-bit counts |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-10-00-000 |

Description

This parameter reports the position FB1.P (→ p. 480) that was saved at the last power-down. The position can be represented in internal counts or in custom units depending on the value of FB1.PUNIT. In internal units, the format is 32:32. The upper 32 bits represent turns, and the lower 32 bits represent position within a turn.

Related Topics

Initial Position Check | FB1.INITPSTATUS (→ p. 469) | FB1.INITPWINDOW (→ p. 470)

19.13.13 FB1.INITPSTATUS

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads result of initial position comparison. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-10-00-000 |

Description

This parameter reports the status of the initial position comparison:

| Value | Description |
|-------|---|
| 0 | Initial position check is in progress. |
| 1 | Initial position check is disabled. |
| 2 | Position difference between power-up and power-down is less than value of FB1.INITPWINDOW at start up. |
| 3 | Position difference between power-up and power-down is greater than or equal to value of FB1.INITPWINDOW at start up. |
| 4 | Initial position check encountered an error condition. |

Related Topics

Initial Position Check | FB1.INITPSAVED (→ p. 468) | FB1.INITPWINDOW (→ p. 470)

19.13.14 FB1.INITPWINDOW

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Reads and writes size of the initial position comparison window. |
| Units | Depends on FB1.PUNIT (→ p. 489). |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-10-00-000 |

Description

This parameter defines the amount by which the position measured at power-up may differ from the position saved at power-down. If the difference is less than the window size, the position change at power off is considered small, and homing may be avoided. If the difference is greater than or equal to the window size, the position change during power off is large enough to require homing. Changes in the window size take effect after a non-volatile memory save and a power cycle. A window value of 0 disables the initial position check.

Related Topics

Initial Position Check | FB1.INITPSTATUS (→ p. 468) | FB1.INITPSTATUS (→ p. 469)

19.13.15 FB1.INITSIGNED

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets initial feedback value as signed or unsigned. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 1 |
| Data Type | Integer |
| See Also | FB1.ORIGIN |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 508Fh/0 | M_01-00-00-000 |

Description

This parameter sets whether the initial value of the feedback read from the feedback device will be set as a signed or as an unsigned value.

0 = Unsigned

1 = Signed

The drive internal process for the feedback initialization is as follows:

1. Reads the position feedback.
2. Adds the origin to the feedback.
3. Determines modulo from Step 2 by the actual feedback bits.
4. Sets the position feedback sign according to FB1.INITSIGNED.

Related Topics

Feedback 1 (→ p. 54)

19.13.16 FB1.LASTIDENTIFIED

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Stores the feedback type auto-detected during the last feedback initialization. Auto-detection will look for this type first. |
| Units | N/A |
| Range | 30 (EnDat 2.1), 32 (BiSS with Sine Cosine), 33 (HIPERFACE), 41 (SFD), 45 (SFD3) |
| Default Value | 41 (SFD) |
| Data Type | Integer |
| Start Version | M_01-10-01-000 |

Description

If FB1.SELECT (→ p. 492) is set to -1, AKD PDMM automatically detects the feedback device. When auto-detection is successful, the identified feedback type is stored in FB1.LASTIDENTIFIED. On the next feedback initialization, the auto-detection algorithm will first look for the last identified type, speeding up feedback initialization and shortening drive boot time.

The time to automatically identify the device is shortened when FB1.LASTIDENTIFIED is already set to the correct feedback type.

Note that this parameter must be saved to non-volatile memory with DRV.NVSAVE (→ p. 428) in order to take effect each time the drive powers up.

Related Topics

Feedback 1 (→ p. 54) | FB1.SELECT (→ p. 492) | FB1.IDENTIFIED (→ p. 467)

19.13.17 FB1.MECHPOS

| General Information | |
|---------------------|--------------------------------|
| Type | R/O Parameter |
| Description | Reads the mechanical position. |
| Units | counts |
| Range | 0 to 4,294,967,295 counts |
| Default Value | N/A |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 35D2h/0 | M_01-00-00-000 |

Description

FB1.MECHPOS reads the mechanical angle which is equal to the lower 32 bits in the 64-bit position feedback word.

Related Topics

Feedback 1 (→ p. 54)

19.13.18 FB1.MEMVER

| General Information | |
|---------------------|--------------------------------------|
| Type | R/O Parameter |
| Description | Returns the memory feedback version. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

FB1.MEMVER returns the memory feedback version (only applicable for feedbacks with memory).

Related Topics

Feedback 1 (→ p. 54)

19.13.19 FB1.MOTORPHASE

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the motor phase of the motor associated with FB1. |
| Units | Electrical degrees |
| Range | 0 to 359 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-11-04-000 |

Description

This parameter sets the phase offset for commutation of the motor associated with FB1. When multiple motors are driven by a single AKD PDMM, this allows different motor configurations to be saved for each value of IL.FBSOURCE (→ p. 556).

When IL.FBSOURCE is set to 0, the value of FB1.MOTORPHASE will overwrite the value of MOTOR.PHASE (→ p. 614). Additionally, while IL.FBSOURCE = 0, a write to MOTOR.PHASE will also overwrite FB1.MOTORPHASE.

Related Topics

Motor (→ p. 50)

19.13.20 FB1.MOTORPOLES

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the number of motor poles associated with FB1. |
| Units | N/A |
| Range | 0 to 510 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-11-04-000 |

Description

This parameter sets the number of motor poles associated with FB1. When multiple motors are driven by a single AKD PDMM, this allows different motor configurations to be saved for each value of IL.FBSOURCE (→ p. 556).

When IL.FBSOURCE is set to 0, the value of FB1.MOTORPOLES will overwrite the current value of MOTOR.POLES (→ p. 618). Additionally, while IL.FBSOURCE = 0, a write to MOTOR.POLES will also overwrite FB1.MOTORPOLES.

Related Topics

Motor (→ p. 50)

19.13.21 FB1.OFFSET

| General Information | |
|---------------------|---|
| Type | NV-Parameter |
| Description | Sets position feedback offset. |
| Units | Rotary: counts, rad, deg, 16-bit counts Linear: counts, mm, μm , 16-bit counts |
| Range | Rotary: -5,123,372,000,000,005.000 to 5,123,372,000,000,005.000 counts -7495.067 to 7495.067 rad -429,436.096 to 429,436.096 deg -78,176,452.637 to 78,176,452.636 16-bit counts Linear: -5,123,372,000,000,005.000 to 5,123,372,000,000,005.000 counts -1192.878*MOTOR.PITCH (\rightarrow p. 617) to 1192.878*MOTOR.PITCH mm -1192877.952*MOTOR.PITCH to 1192877.952*MOTOR.PITCH μm -78176452.637 to 78176452.636 16-bit counts |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3490h/0 | M_01-00-00-000 |

Description

FB1.OFFSET is a value added to the position feedback (PL.FB (\rightarrow p. 644)).

Example

If PL.FB is 10 deg and FB1.OFFSET is set to -10 deg, then the next read of PL.FB will return ~0 deg.

Related Topics

Feedback 1 (\rightarrow p. 54)| PL.FB (\rightarrow p. 644)

19.13.22 FB1.ORIGIN

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Adds to the initial feedback position. |
| Units | Depends on UNIT.PROTARY or UNIT.PLINEAR Rotary: counts, rad, deg, custom units, 16-bit counts Linear: counts, mm, µm, custom units, 16-bit counts |
| Range | Rotary: 0.000 to 5,123,372,000,000.000 counts 0.000 to 7,495.067 rad 0.000 to 429,436.096 deg 0.000 to 5,964.390 custom units 0.000 to 78,176,452.636 16-bit counts Linear: 0.000 to 5,123,372,000,000.000 counts 0.000 to 1,192.878 mm 0.000 to 1,192,877.952 µm 0.000 to 5,964.390 custom units 0.000 to 78,176,452.636 counts 16 Bit |
| Default Value | 0 counts |
| Data Type | Float |
| See Also | FB1.INITSIGNED |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3656h/0 | M_01-00-00-000 |

Description

FB1.ORIGIN is a value that is added to the feedback device position.

If the modulo is disabled (PL.MODPEN is set to disabled), then the origin is added to the position feedback.

Initial position value = (<feedback from device> + FB1.ORIGIN)

However, if modulo is enabled (PL.MODPEN is set to enabled) then the modulo is applied to both the feedback device position and the origin value.

The modulo value is defined by PL.MODP1 and PL.MODP2

Initial position value = (<feedback from device> + FB1.ORIGIN) modulo

The drive internal process for the feedback initialization is as follows:

1. Reads the position feedback.
2. Adds the origin to the feedback.
3. Determines modulo from Step 2 by the actual feedback bits.
4. Sets the position feedback sign according to FB1.INITSIGNED (→ p. 471).

Example

This example assumes that the drive is connected to a single turn feedback device with memory.

FB1.ORIGIN is set to 22 degrees and saved into NV memory.

Drive boots and reads from feedback device position 340 degrees. According to the description section above, calculation will be:

With PL.MODPEN disabled:

Initial feed back will be set to 366 degrees.

$$(340 + 22) = 366 \text{ degrees.}$$

With PL.MODPEN enabled:

Initial feedback will be set to 2 degrees.

$$\text{PL.MODP1} = 0$$

$$\text{PL.MODP2} = 360$$

$$(340 + 22) \text{ modulo } 360 = 2 \text{ degrees.}$$

Related Topics

7.2 Feedback 1

19.13.23 FB1.P

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads position from the primary feedback. |
| Units | Depends on FB1.PUNIT (→ p. 489) counts or custom units. |
| Range | N/A |
| Default Value | N/A |
| Data Type | S64 |
| Start Version | M_01-05-08-000 |

Description

This parameter reads the position of the primary feedback device connected to X10. The position can be read as counts or in custom units. This is the raw position read back from the device. The output format is 32:32, the upper 32 bits represent the multi-turns and the lower 32 bits represent the position of the feedback.

Related Topics

Feedback 1 (→ p. 54) | FB1.PUNIT (→ p. 489) | FB1.PIN | FB1.POUT

19.13.24 FB1.PDIR

| General Information | |
|---------------------|---|
| Type | NV-Parameter |
| Description | Sets the counting direction for feedback channel 1. |
| Units | None |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | U8 |
| See Also | N/A |
| Start Version | M_01-05-11-000 |

Description

FB1.PDIR will change the sign and with it the direction of feedback channel 1.

Example

If position feedback = 35,185.932 counts and you set:

--> FB1.PDIR 1

then position feedback = -35,185.932 counts

19.13.25 FB1.PFIND

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | A procedure that allows the user to find the commutation angle for encoder feedback, which has no halls. |
| Units | NA |
| Range | 0, 1 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | FB1.PFINDCMDU (→ p. 483) |
| Start Version | M_01-00-00-000 |

Description

A procedure that allows the user to find the commutation angle for encoder feedback (which has no Halls).

Related Topics

Feedback 1 (→ p. 54)

19.13.26 FB1.PFINDCMDU

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Current value used during the phase finding procedure (PFB.PFIND=1) |
| Units | A |
| Range | 0 to DRV.IPEAK |
| Default Value | 0 |
| Data Type | Float |
| See Also | PFB.PFIND |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5096h/0 | M_01-00-00-000 |

Description

FB1.PFINDCMDU sets the current value used during the phase finding procedure.

Related Topics

Feedback 1 (→ p. 54)

19.13.27 FB1.PIN

| General Information | |
|---------------------|------------------------------------|
| Type | NV Parameter |
| Description | Sets gear IN for FB1.P (→ p. 480). |
| Units | N/A |
| Range | 1 to 4,294,967,295 |
| Default Value | 100 |
| Data Type | Integer |
| Start Version | M_01-05-011-000 |

Description

Use FB1.PIN with FB1.POUT (→ p. 487) to set the user units for FB1.P (→ p. 480).

Related Topics

Feedback 1 (→ p. 54) | FB1.PUNIT (→ p. 489) | FB1.P (→ p. 480) | FB1.POUT (→ p. 487)

19.13.28 FB1.POFFSET

| General Information | |
|---------------------|--|
| Type | NV-Parameter |
| Description | Sets the offset for primary feedback. |
| Units | counts, custom units |
| Range | -5,123,372,000,000,005.000 to 5,123,372,000,000,005.000 counts or -10,485,760.000 to 10,485,760.000 custom units |
| Default Value | 0 |
| Data Type | S64 |
| See Also | N/A |
| Start Version | M_01-05-11-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3490h/0 | M_01-00-00-000 |

Description

FB1.POFFSET is the value added to the primary feedback position (FB1.P (→ p. 480)).

Example

If FB1.P is 10000 counts and FB1.POFFSET is set to -10000 counts, then the next read of FB1.P will return ~0 counts.

19.13.29 FB1.POLES

| General Information | |
|---------------------|------------------------------------|
| Type | R/W Parameter |
| Description | Sets the number of feedback poles. |
| Units | N/A |
| Range | 2 to 510 |
| Default Value | 2 |
| Data Type | Integer |
| See Also | MOTOR.POLES |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5097h/0 | M_01-00-00-000 |

Description

FB1.POLES sets the number of individual poles in the feedback device. This variable is used for the commutation function, as well as for velocity feedback scaling, and represents the number of individual poles (not pole pairs). The division value of motor poles (MOTOR.POLES) and feedback poles (FB1.POLES) must be an integer when moving drive to enable, otherwise a fault is issued.

Related Topics

Feedback 1 (→ p. 54)

19.13.30 FB1.POUT

| General Information | |
|---------------------|------------------------------------|
| Type | NV Parameter |
| Description | Sets gear IN for FB1.P (→ p. 480). |
| Units | N/A |
| Range | 1 to 4,294,967,295 |
| Default Value | 20 |
| Data Type | Integer |
| Start Version | M_01-05-011-000 |

Description

Use FB1.POUT with FB1.PIN (→ p. 484) to set the user units for FB1.P (→ p. 480).

Related Topics

Feedback 1 (→ p. 54) | FB1.PUNIT (→ p. 489) | FB1.P (→ p. 480) | FB1.PIN (→ p. 484)

19.13.31 FB1.PSCALE

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets position scaling value for fieldbus transferred position objects. |
| Units | N/A |
| Range | 0 to 32 |
| Default Value | 20 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

Position values transferred over fieldbus are converted from native 64-bit values to a maximum 32-bit position value. This parameter sets the resolution/revolution of position values back to the controller.

FB1.PSCALE determines the counts per revolution of position values delivered by fieldbus. The default value is 20, which yields 2^{20} counts/revolution. This scaling is used for CAN PDOs 6064 (Position Actual Value) and 60F4 (Following Error Actual Value).

Example

The drive always works internally with 64-bit position values. The drive internal 64-bit actual position should contain the following value:

0x0000.0023.1234.ABCD

The lower 32 bits represent the mechanical angle of the feedback. The upper 32 bits represent the number of turns.

FB1.PSCALE = 20

The 32-bit position is: 0x0231234A

FB1.PSCALE = 16

The 32-bit position is: 0x00231234

Related Topics

Feedback 1 (→ p. 54)

19.13.32 FB1.PUNIT

| General Information | |
|---------------------|--------------------------|
| Type | NV Parameter |
| Description | Sets the unit for FB1.P. |
| Units | N/A |
| Range | 0, 3 |
| Default Value | 0 |
| Data Type | U8 |
| See Also | N/A |
| Start Version | M_01-05-11-000 |

Description

FB1.PUNIT sets the position unit for FB1.P.

| Value | Description |
|-------|------------------------------------|
| 0 | Counts (32.32 format) |
| 3 | (FB1.PIN/FB1.POUT) per revolution. |

Related Topics

FB1.P (→ p. 480)

19.13.33 FB1.RESKTR

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the resolver nominal transformation ratio. |
| Units | N/A |
| Range | 0.001 to 50.000 |
| Default Value | 0.5 |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5099h/0 | M_01-00-00-000 |

Description

This parameter sets the resolver nominal transformation ratio. It affects the resolver excitation output amplitude.

The value can be obtained from the resolver data sheet.

Related Topics

Feedback 1 (→ p. 54)

19.13.34 FB1.RESREFPHASE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the electrical degrees of phase lag in the resolver. |
| Units | electrical degrees |
| Range | -180 to 180° |
| Default Value | -2° |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 509Ah/0 | M_01-00-00-000 |

Description

This parameter sets the electrical degrees of phase lag in the resolver.

See the motor resolver datasheet for the value for this parameter .

Related Topics

Feedback 1 (→ p. 54)

19.13.35 FB1.SELECT

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets user entered type or identified type (-1). |
| Units | N/A |
| Range | -1, 1, 10, 11, 12, 20, 21, 30, 31, 32, 33, 34, 40, 41, 42, 43, 45, 46, 50 |
| Default Value | -1 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 353Bh/0 | M_01-00-00-000 |

Description

FB1.SELECT sets the feedback type manually (see FB1.IDENTIFIED) or allows the drive to automatically identify the feedback type on power up.

FB1 5V Power Supply Notes:

If FB1.SELECT = -1, the 5V power supply will only turn on during the initialization phase. If the feedback type cannot be automatically detected, the 5V power supply will be turned off, and remain off until a DRV.CLRFAULTS is issued (which will re-start the FB1 initialization), or FB1.SELECT is manually set to a specific feedback type.

When FB1.SELECT is manually set to a specific feedback type, the FB1 5V power supply will remain on, unless a short is detected. This is useful for encoder types that may require calibration.

FB1.SELECT Input Values

Changing these input values will manually set the value of FB1.IDENTIFIED (→ p. 467).

NOTE

If the feedback setting fails, FB1.IDENTIFIED is automatically set to 0 (no feedback identified).

| Input Value | Description |
|-------------|--|
| -1 | The drive automatically identifies the type of feedback as part of the power up process. Setting this value does not modify FB1.IDENTIFIED, unless it is saved in the NV memory for the next power up. If a feedback with memory is connected to the drive, the value of FB1.IDENTIFIED is set automatically to the feedback identified and all parameters read from the feedback are set according to the values read from the feedback. If no feedback is connected or a feedback with no memory is connected, the value of FB1.IDENTIFIED is set to 0 (no feedback identified) and all values normally read from the feedback are read from NV memory (if stored in NV) otherwise they are set to the default values. |
| 0 | Unknown. No feedback identified. |
| 1 | Manually sets the type to No Encoder. This setting can be used if no feedback device is connected to the X10 connector. This may be the case if FB2 or FB3 is used for position or velocity control. |
| 10 | Manually sets the type to Incremental encoder with A/B Quad, marker pulse and Hall. |
| 11 | Manually sets the type to Incremental encoder with A/B Quad, marker pulse and no Hall. |

| Input Value | Description |
|-------------|---|
| 12 | Manually sets the type to Halls Only feedback (added 01-13-08-000). This feedback type only supports operating in Torque mode and Velocity mode. Due to heavy quantization, large current spikes occur at hall transition points, it is advisable to tune the observer to a low frequency (like 10Hz) and set the observer gain appropriately to achieve the best operating results. |
| 20 | Manually sets the type to Sine Encoder , with marker pulse and Hall. |
| 21 | Manually sets the type to Sine encoder , with marker pulse & No Halls. |
| 30 | Manually sets the type to Endat 2.1 with Sine Cosine. |
| 31 | Manually sets the type to Endat 2.2. |
| 32 | Manually sets the type to BiSS (Analog) with Sine Cosine. |
| 33 | Manually sets the type to Hiperface. Note that all Hiperface feedback types are supported by the AKD PDMM. This includes SEL/SEK 37, SEL/SEK 52, SKM/SKS 36, SRS/SRM 50, SRS/SRM 60, SEK 90, SEK160, and SEK 260. The AKD PDMM drive will support any new Hiperface device, since any new device will be released with a label type of 0xFF. Devices with this label type have all of the pertinent information to configure these devices (number of single turn bits, number of multi-turn bits, and number of sine/cosine periods) stored in their memory. The AKD PDMM is able to read this information, and automatically configure the drive for proper operation. Note that the devices SEK 90, SEK 160, and SEK 260 are label type 0xFF. |
| 34 | Manually sets the type to BiSS Mode C, Renishaw. |
| 40 | Manually sets the type to Resolver. |
| 41 | Manually sets the type to SFD (Smart Feedback Device). |
| 42 | Manually sets the type to Tamagawa. This feedback type is not available on models with extension type NA (AKD PDMM Models (→ p. 26)). |
| 43 | Manually sets the type to use Cyclic Feedback from the Motion Network. |
| 45 | Manually sets the type to SFD3. This value is only supported on model numbers AKD PDMM-xxyyzz-NBxx. This value is only supported for firmware version 01-11-00-000 and later. To use 2 wire SFD3 there must be an external feedback connection adapter between the 2 wires and the AKD X10 primary feedback connector. The feedback connection adapter comes standard on Kollmorgen single motor cables. |
| 46 | Manually sets the type to Hiperface DSL. This value is only supported on model numbers AKD PDMM-xxyyzz-NBxx. This value is only supported for firmware versions 01-09-00-000 and later. To use 2 wire Hiperface DSL there must be an external feedback connection adapter between the 2 wires and the AKD X10 primary feedback connector. The feedback connection adapter comes standard on Kollmorgen single motor cables. |
| 50 | This value is reserved and should not be used. |
| 63 | Manually sets the type to a simulated motor. This value is only supported on model numbers AKD PDMM-xxyyzz-NBxx. |

Related Topics

7.2 Feedback 1 | FB1.IDENTIFIED (→ p. 467)

19.13.36 FB1.TRACKINGCAL

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Controls tracking calibration algorithm. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 509Ch/0 | M_01-00-00-000 |

Description

This parameter turns the tracking calibration algorithm for 1 Vp-p analog sine-cosine feedback on or off. When enabled and the axis is moving, the interface nulls any offset voltage, corrects gain mismatches between the two channels, and corrects the two channels to be exactly 90 Deg apart. This process automatically turns on above 48 kHz sine-cosine input frequency and when going below 47 kHz the offset, gain, and quadrature adjustment values are sampled and held. This process makes the motion smoother while moving with less unintended acoustic noise. However, because the adjustments will slightly change move-to-move, the repeatability of position holds is slightly degraded.

0 = Tracking calibration is off.

1 = Tracking calibration is on.

Related Topics

Feedback 1 (→ p. 54)

19.13.37 FB1.USERBYTE0 to FB1.USERBYTE7

| General Information | |
|---------------------|---|
| Type | R/W |
| Description | Reads and writes data stored in eight 8 bit bytes in an Endat or Hiperface feedback device. |
| Units | N/A |
| Range | 0 to 255 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-05-08-000 |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|---------------|----------------------|
| EtherCAT COE and CANopen | 34B2h/1 | FB1.USERBYTE1 | M_01-00-00-000 |
| | 34B2h/2 | FB1.USERBYTE2 | |
| | 34B2h/3 | FB1.USERBYTE3 | |
| | 34B2h/4 | FB1.USERBYTE4 | |
| | 34B2h/5 | FB1.USERBYTE5 | |
| | 34B2h/6 | FB1.USERBYTE6 | |
| | 34B2h/7 | FB1.USERBYTE7 | |
| | 34B2h/8 | FB1.USERBYTE8 | |

Description

FB1.USERBYTE, FB1.USERWORD, and FB1.USERDWORD share two 32 bit words in an Endat or Hiperface feedback device. These three parameters offer different ways to store and access the data stored in these two 32 bit words. The table below defines each parameters function. The values are automatically stored non-volatile in the feedback device.

| Parameter | Bits | Attributes |
|----------------------------------|--------|--------------------------|
| FB1.USERBYTE0 to FB1.USERBYTE7 | 8 bit | signed or unsigned char |
| FB1.USERWORD0 to FB1.USERWORD3 | 16 bit | signed or unsigned short |
| FB1.USERDWORD0 to FB1.USERDWORD1 | 32 bit | singed or unsigned int |

The parameters overlap each other in the following way:

| | | | | | | | |
|--------|-------|-------|-------|--------|-------|-------|------|
| BYTE0 | BYTE1 | BYTE2 | BYTE3 | BYTE4 | BYTE5 | BYTE6 | BYT7 |
| WORD0 | | WORD1 | | WORD2 | | WORD3 | |
| DWORD0 | | | | DWORD1 | | | |

For example, if BYTE1 is modified, WORD0 and DWORD0 are modified as well.

Example

```
-->FB1.USERDWORD1 65536
-->FB1.USERBYTE1
0
-->FB1.USERBYTE2
```

```
1
-->FB1.USERBYTE3
0

-->FB1.USERBYTE3 1 (write to the highest byte of FB1.USERDWORD0)
-->FB1.USERDWORD0
16842752
-->FB1.USERWORD0
0
-->FB1.USERWORD1
257
```

Related Topics

Feedback 1 (→ p. 54) | FB1.USERWORD0 to FB1.USERWORD3 (→ p. 499) | FB1.USERDWORD0 to FB1.USERDWORD1 (→ p. 497)

19.13.38 FB1.USERDWORD0 to FB1.USERDWORD1

| General Information | |
|---------------------|--|
| Type | R/W |
| Description | Reads and writes data stored in two 32 bit words in an Endat or Hiperface feedback device. |
| Units | N/A |
| Range | 0 to 4,294,967,295 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-05-08-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|--------------------|----------------------|
| EtherCAT COE and CANopen | 34B0h (→ p. 986)/1 | FB1.USERDWORD1 |
| | 34B0h/2 | FB1.USERDWORD2 |
| | | M_01-00-00-000 |

Description

FB1.USERBYTE, FB1.USERWORD, and FB1.USERDWORD share two 32 bit words in an Endat or Hiperface feedback device. These three parameters offer different ways to store and access the data stored in these two 32 bit words. The table below defines each parameters function. The values are automatically stored non-volatile in the feedback device.

| Parameter | Bits | Attributes |
|----------------------------------|--------|--------------------------|
| FB1.USERBYTE0 to FB1.USERBYTE7 | 8 bit | signed or unsigned char |
| FB1.USERWORD0 to FB1.USERWORD3 | 16 bit | signed or unsigned short |
| FB1.USERDWORD0 to FB1.USERDWORD1 | 32 bit | singed or unsigned int |

The parameters overlap each other in the following way:

| | | | | | | | |
|--------|-------|-------|-------|--------|-------|-------|------|
| BYTE0 | BYTE1 | BYTE2 | BYTE3 | BYTE4 | BYTE5 | BYTE6 | BYT7 |
| WORD0 | | WORD1 | | WORD2 | | WORD3 | |
| DWORD0 | | | | DWORD1 | | | |

For example, if BYTE1 is modified, WORD0 and DWORD0 are modified as well.

Example

```
-->FB1.USERDWORD1 65536
-->FB1.USERBYTE1
0
-->FB1.USERBYTE2
1
-->FB1.USERBYTE3
0

-->FB1.USERBYTE3 1 (write to the highest byte of FB1.USERDWORD0)
-->FB1.USERDWORD0
```

16842752

-->FB1.USERWORD0

0

-->FB1.USERWORD1

257

Related Topics

Feedback 1 (→ p. 54) | FB1.USERBYTE0 to FB1.USERBYTE7 (→ p. 495) | FB1.USERWORD0 to FB1.USERWORD3 (→ p. 499)

19.13.39 FB1.USERWORD0 to FB1.USERWORD3

| General Information | |
|---------------------|---|
| Type | R/W |
| Description | Reads and writes data stored in four 16 bit words in an Endat or Hiperface feedback device. |
| Units | N/A |
| Range | 0 to 65,535 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-05-08-000 |

| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|---------------|----------------------|
| EtherCAT COE and CANopen | 34B1h/1 | FB1.USERWORD1 | M_01-00-00-000 |
| | 34B1h/2 | FB1.USERWORD2 | |
| | 34B1h/3 | FB1.USERWORD3 | |
| | 34B1h/4 | FB1.USERWORD4 | |

Description

FB1.USERBYTE, FB1.USERWORD, and FB1.USERDWORD share two 32 bit words in an Endat or Hiperface feedback device. These three parameters offer different ways to store and access the data stored in these two 32 bit words. The table below defines each parameters function. The values are automatically stored non-volatile in the feedback device.

| Parameter | Bits | Attributes |
|----------------------------------|--------|--------------------------|
| FB1.USERBYTE0 to FB1.USERBYTE7 | 8 bit | signed or unsigned char |
| FB1.USERWORD0 to FB1.USERWORD3 | 16 bit | signed or unsigned short |
| FB1.USERDWORD0 to FB1.USERDWORD1 | 32 bit | signed or unsigned int |

The parameters overlap each other in the following way:

| | | | | | | | |
|--------|-------|-------|-------|--------|-------|-------|------|
| BYTE0 | BYTE1 | BYTE2 | BYTE3 | BYTE4 | BYTE5 | BYTE6 | BYT7 |
| WORD0 | | WORD1 | | WORD2 | | WORD3 | |
| DWORD0 | | | | DWORD1 | | | |

For example, if BYTE1 is modified, WORD0 and DWORD0 are modified as well.

Example

```
-->FB1.USERDWORD1 65536
-->FB1.USERBYTE1
0
-->FB1.USERBYTE2
1
-->FB1.USERBYTE3
0
```

```
-->FB1.USERBYTE3 1 (write to the highest byte of FB1.USERDWORD0)
-->FB1.USERDWORD0
16842752
-->FB1.USERWORD0
0
-->FB1.USERWORD1
257
```

Related Topics

Feedback 1 (→ p. 54) | FB1.USERBYTE0 to FB1.USERBYTE7 (→ p. 495) | FB1.USERDWORD0 to FB1.USERDWORD1 (→ p. 497)

19.14 FB3 Parameters

This section describes the FB3 parameters.

| | |
|-------------------------------------|------------|
| 19.14.1 FB3.DIR | 502 |
| 19.14.2 FB3.ENCRES | 503 |
| 19.14.3 FB3.MODE | 504 |
| 19.14.4 FB3.MOTORPHASE | 505 |
| 19.14.5 FB3.MOTORPOLES | 506 |
| 19.14.6 FB3.P | 507 |
| 19.14.7 FB3.PDIR | 508 |
| 19.14.8 FB3.PIN | 509 |
| 19.14.9 FB3.POFFSET | 510 |
| 19.14.10 FB3.POUT | 511 |
| 19.14.11 FB3.PUNIT | 512 |

19.14.1 FB3.DIR

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the counting direction for the position feedback when PL.FBSOURCE = 2. |
| Units | None |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | U8 |
| See Also | N/A |
| Start Version | M_01-11-03-000 |

Description

FB3.DIR will change the sign and with it the direction of the position feedback (PL.FB (→ p. 644)) when PL.FBSOURCE = 2 (FB3).

19.14.2 FB3.ENCRES

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the resolution of the motor encoder for feedback 3. |
| Units | None |
| Range | 0 to 4,294,967,295 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-11-03-000 |

Description

Feedback 3 is natively scaled to 4,294,967,296 counts per revolution for rotary motors, and 4,294,967,296 per motor pole pitch (MOTOR.PITCH (→ p. 617)) for linear motors. When FB3.ENCRES is non-zero, scaling is applied on top of the native scaling.

Rotary Example:

A second feedback device is placed at the load which has a 1:4 gear ratio (motor does four revolutions per secondary revolution). In this case, FB3.ENCRES should be $4,294,967,296 / 4 = 1,073,741,824$

Linear Example:

Linear encoder resolution can be calculated as the number of encoder counts per one revolution of the motor. For example, if a linear setup has a 10nm scale (100,000 counts/nm), and the ballscrew has a 10mm lead with no gearbox, it will result in 1,000,000 counts for every revolution of the motor shaft. For this example, FB3.ENCRES should be set to 1,000,000.

19.14.3 FB3.MODE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Selects the type of feedback connected to X9. |
| Units | N/A |
| Range | 0 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | NA |
| Start Version | M_01-04-15-000 |

Description

This parameter selects the type of feedback connected to X9. The position is reported as the tertiary feedback position, by FB3.P.

| Value | Feedback |
|-------|---------------------------|
| 0 | Endat 2.2 Feedback Device |

This parameter is only supported on drives with model numbers similar to AKD PDMM-x-xxxxx-NBxx-xxxx.

19.14.4 FB3.MOTORPHASE

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the motor phase of the motor associated with FB3. |
| Units | Electrical degrees |
| Range | 0 to 359 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-11-04-000 |

Description

This parameter sets the phase offset for commutation of the motor associated with FB3. When multiple motors are driven by a single AKD PDMM, this allows different motor configurations to be saved for each value of IL.FBSOURCE (→ p. 556).

When IL.FBSOURCE is set to 2, the value of FB3.MOTORPHASE will overwrite the value of MOTOR.PHASE (→ p. 614). Additionally, while IL.FBSOURCE = 2, a write to MOTOR.PHASE will also overwrite FB3.MOTORPHASE.

Related Topics

Motor (→ p. 50)

19.14.5 FB3.MOTORPOLES

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the number of motor poles associated with FB3. |
| Units | N/A |
| Range | 0 to 510 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-11-04-000 |

Description

This parameter sets the number of motor poles associated with FB3. When multiple motors are driven by a single AKD PDMM, this allows different motor configurations to be saved for each value of IL.FBSOURCE (→ p. 556).

When IL.FBSOURCE is set to 2, the value of FB3.MOTORPOLES will overwrite the current value of MOTOR.POLES (→ p. 618). Additionally, while IL.FBSOURCE = 2, a write to MOTOR.POLES will also overwrite FB3.MOTORPOLES.

Related Topics

Motor (→ p. 50)

19.14.6 FB3.P

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads position from the tertiary feedback. |
| Units | Depends on FB3.PUNIT (→ p. 512) counts or custom units. |
| Range | N/A |
| Default Value | N/A |
| Data Type | S64 |
| See Also | N/A |
| Start Version | M_01-04-15-000 |

Description

This parameter reads the position back from the tertiary feedback device that is connected to X9, when `DRV.EMUEMODE = 11`. The position can be read as 64-bit signed count or in customer units. Values read by this parameter depend on `FB3.Dir` and `FB3.OFFSET`.

This parameter is only supported on drives with model numbers similar to AKD PDMM-x-xyzz-NBxx-yyzz and will only work with Endat 2.2 multiturn encoder. The output format is 32:32 the upper 32 bits represents the multiturns and the lower 32 bits for position of the feedback.

Related Topics

`DRV.EMUEMODE` (→ p. 392) | `FB3.PUNIT` (→ p. 512) | `FB3.PIN` | `FB3.POUT` | `FB3.PDIR` (→ p. 508) |

19.14.7 FB3.PDIR

| General Information | |
|---------------------|---|
| Type | NV-Parameter |
| Description | Sets the counting direction for feedback channel 3. |
| Units | None |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | U8 |
| See Also | N/A |
| Start Version | M_01-05-11-000 |

Description

FB3.PDIR will change the sign and with it the direction of feedback channel 3 (FB3.P (→ p. 507)).

Example

If FB3.P = 35,185.932 and you set:

→ FB3.PDIR 1

then FB3.P = -35,185.932

19.14.8 FB3.PIN

| General Information | |
|---------------------|------------------------------------|
| Type | NV Parameter |
| Description | Sets gear IN for FB3.P (→ p. 507). |
| Units | N/A |
| Range | 1 to 4,294,967,295 |
| Default Value | 100 |
| Data Type | Integer |
| Start Version | M_01-05-011-000 |

Description

Use FB3.PIN with FB3.POUT (→ p. 511) to set the user units for FB3.P (→ p. 507).

Related Topics

Feedback 1 (→ p. 54) | FB3.PUNIT (→ p. 512) | FB3.P (→ p. 507) | FB3.POUT (→ p. 511)

19.14.9 FB3.POFFSET

| General Information | |
|---------------------|--|
| Type | NV-Parameter |
| Description | Sets the offset for tertiary feedback. |
| Units | counts, custom units |
| Range | -5,123,372,000,000,005.000 to 5,123,372,000,000,005.000 counts or -10,485,760.000 to 10,485,760.000 custom units |
| Default Value | 0 |
| Data Type | S64 |
| See Also | N/A |
| Start Version | M_01-05-11-000 |

Description

FB3.POFFSET is the value added to the primary feedback position (FB3.P (→ p. 507)).

Example

If FB3.P is 10000 counts and FB3.POFFSET is set to -10000 counts, then the next read of FB3.P will return ~0 counts.

19.14.10 FB3.POUT

| General Information | |
|---------------------|------------------------------------|
| Type | NV Parameter |
| Description | Sets gear IN for FB3.P (→ p. 507). |
| Units | N/A |
| Range | 1 to 4,294,967,295 |
| Default Value | 20 |
| Data Type | Integer |
| Start Version | M_01-05-011-000 |

Description

Use FB3.POUT with FB3.PIN (→ p. 509) to set the user units for FB3.P (→ p. 507).

Related Topics

Feedback 1 (→ p. 54) | FB3.PUNIT (→ p. 512) | FB3.P (→ p. 507) | FB3.PIN (→ p. 509)

19.14.11 FB3.PUNIT

| General Information | |
|---------------------|--------------------------|
| Type | NV Parameter |
| Description | Sets the unit for FB3.P. |
| Units | N/A |
| Range | 0, 3 |
| Default Value | 0 |
| Data Type | U8 |
| See Also | N/A |
| Start Version | M_01-05-11-000 |

Description

FB3.UNIT sets the position unit for FB3.P.

| Value | Description |
|-------|------------------------------------|
| 0 | Counts (32.32 format) |
| 3 | (FB3.PIN/FB3.POUT) per revolution. |

Related Topics

FB3.P (→ p. 507)

19.15 FBUS Parameters

This section describes the FBUS parameters.

| | |
|--|------------|
| 19.15.1 FBUS.PARAM1 TO FBUS.PARAM10 | 514 |
| 19.15.2 FBUS.PLLSTATE | 517 |
| 19.15.3 FBUS.PLLTHRESH | 518 |
| 19.15.4 FBUS.PROTECTION | 519 |
| 19.15.5 FBUS.REMOTE | 521 |
| 19.15.6 FBUS.SAMPLEPERIOD | 522 |
| 19.15.7 FBUS.STATE | 523 |
| 19.15.8 FBUS.SYNACQUIREWND | 524 |
| 19.15.9 FBUS.SYNCACT | 525 |
| 19.15.10 FBUS.SYNCDIST | 526 |
| 19.15.11 FBUS.SYNCKLOCKWND | 527 |
| 19.15.12 FBUS.SYNCWND | 528 |
| 19.15.13 FBUS.TYPE | 529 |

19.15.1 FBUS.PARAM1 TO FBUS.PARAM10

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Set fieldbus specific meanings. |
| Units | N/A |
| Range | See table below. |
| Default Value | See table below. |
| Data Type | Unsigned 32 |
| See Also | CANbus Communication Manual, EtherCAT Communication Manual |
| Start Version | M_01-00-00-000 |

| Parameter | EtherCAT COE Range | CANopen Range |
|--------------|---|---------------------|
| FBUS.PARAM01 | AKD PDMM-C: 0 to 65,535 Other AKD PDMMs: N/A | 125; 250; 500; 1000 |
| FBUS.PARAM02 | 0 to 1 | 0 to 1 |
| FBUS.PARAM03 | 0 to 65,535 | N/A |
| FBUS.PARAM04 | 0 to 1 | 0 to 1 |
| FBUS.PARAM05 | 0 to 511 | 0 to 511 |

| Parameter | EtherCAT COE Default Value | CANopen Default Value |
|--------------|--|-----------------------|
| FBUS.PARAM01 | AKD PDMM-C: 0 Other AKD PDMMs: N/A | 125 |
| FBUS.PARAM02 | 1 | 0 |
| FBUS.PARAM03 | 0 | N/A |
| FBUS.PARAM04 | 1 | 0 |
| FBUS.PARAM05 | 0 | 0 |

| Fieldbus | Index/Subindex | | | | Object Start Version |
|--------------------------|----------------|--------------|---------|--------------|----------------------|
| EtherCAT COE and CANopen | 36E5h/0 | FBUS.PARAM01 | 36EAh/0 | FBUS.PARAM06 | M_01-00-00-000 |
| | 36E6h/0 | FBUS.PARAM02 | 36Ebh/0 | FBUS.PARAM07 | |
| | 36E7h/0 | FBUS.PARAM03 | 36ECh/0 | FBUS.PARAM08 | |
| | 36E8h/0 | FBUS.PARAM04 | 36EDh/0 | FBUS.PARAM09 | |
| | 36E9h/0 | FBUS.PARAM05 | 36EEh/0 | FBUS.PARAM10 | |

Description

FBUS.PARAM01 sets the baud rate for the CANbus. Supported baud rates are 125, 250, 500 and 1000 kBaud. On AKD PDMM-C, FBUS.PARAM01 sets and stores the EtherCAT station alias for the ESC (EtherCAT slave controller) of string 2.

FBUS.PARAM02 switches the phase locked loop (PLL) for synchronized use: 0 = OFF, 1 = ON.
 FBUS.PARAM02 only works when FBUS.TYPE = 3 (CANopen).

FBUS.PARAM03 sets and stores the EtherCAT station alias for the ESC (EtherCAT slave controller). On AKD PDMM-C, FBUS.PARAM03 sets and stores the EtherCAT station alias for the ESC (EtherCAT slave controller) of string 1.

FBUS.PARAM04 switches the surveillance of SYNC-signals: 0 = OFF, 1 = ON.

FBUS.PARAM05 is used to configure some special behaviors of the DS402.

FBUS.PARAM06 - FBUS.PARAM10 are reserved.

FBUS.PARAM04 Additional Notes

FBUS.PARAM04 enables (1) or disables(0) the synchronization supervision of the CAN or EtherCAT field-bus.

Default values:

- CAN: disabled (0)
- EtherCAT: enabled (1)

The synchronization supervision is active when:

- FBUS.PARAM04 = 1
- The first CANopen Sync message or first EtherCAT frame was received.

FBUS.PARAM05 Additional Notes

Bit 0 configures the behavior of DS402 state machine in case of fault resets.

| | | |
|-------|---|--|
| Bit 0 | 1 | Faults can only be reset using DS402 control word bit 7. |
| | 0 | The reset can also be done via telnet or digital input and the DS402 state machine reflects this condition. |
| Bit 1 | 1 | The state of the hardware enable does not change the state machine state Operation Enable. |
| | 0 | If the state Operation Enable or Switched on is active it falls back to the state switched On Disabled, if the Hardware enable goes to 0. |
| Bit 2 | 1 | EWV ¹ /Telnet can not software enable the drive, when CANopen/EtherCAT are Operational. |
| | 0 | EWV/Telnet can software enable the drive. NOTE: During commissioning this bit should be set to 1 to avoid influences to DS402 power stage state machine. The field bus should not be in operation as well to avoid influence to test functions of Workbench. |
| Bit 3 | 1 | DS402-state machine is not influenced, if the software-enable is taken away via Telnet. |
| | 0 | DS402-state machine is influenced, if the software-enable is taken away via Telnet. |
| Bit 4 | 1 | Scaling is done via special DS402 - objects (independent on units) |
| | 0 | Scaling for position, velocity and acceleration objects is done via UNIT parameters. |

¹Embedded Workbench Views

| | | |
|--|---|--|
| Bit 5 (EtherCAT only) | 1 | FBUS.PARAM03 defines the station alias address if not 0. If FBUS.PARAM03 set to 0, the address will be taken from rotary switches instead, if they are not 0. The EtherCAT master has the ability to use the alias address, selected by the drive, or issue its own. |
| | 0 | The rotary switches define the station alias address if not 0. If the rotary switches are set to 0, the address will be taken from FBUS.PARAM03 instead, if it is not 0. |
| Bit 6 | 1 | Bit 0 of parameter MT.CNTL (object 35B9 sub 0) can be accessed. |
| | 0 | Bit 0 of parameter MT.CNTL (object 35B9 sub 0) is exclusively used for DS402 controlword. |
| Bit 7 | 1 | All capture objects (0x20A0-0x20A3, 0x20A6, 0x20A7, 0x60BA to 0x60BD) are scaled like object 0x6063. |
| | 0 | All capture objects (0x20A0-0x20A3, 0x20A6, 0x20A7, 0x60BA to 0x60BD) are scaled like object 0x6064. |
| Bit 8 | 1 | DS402-state SWITCHED ON means power stage disabled. |
| | 0 | DS402-state SWITCHED ON means power stage enabled. |
| Bit 9 | 1 | SDO content of object 0x6063 is the same as PDO content. |
| | 0 | SDO content of object 0x6063 depends on AKD PDMM unit parameters. |
| Bit 10 (Bit 10 is active only, if Bit 8 is set) | 1 | State "Switch On" can be reached without the high-level voltage being active. |
| | 0 | State "Switch On" can only be reached when the high-level voltage is active; otherwise the drive will stay in "Ready to Switch On". |
| Bit 11 | 1 | No emergency messages over CANopen are triggered when a drive warning occurs. |
| | 0 | Emergency messages over CANopen are triggered when a drive warning occurs. |
| Bit 12 | | reserved |
| Bit 13 (EtherCAT only) | 1 | Downloaded parameter file is stored automatically to nonvolatile memory. |
| | 0 | Downloaded parameter file is not stored automatically to nonvolatile memory. |

19.15.2 FBUS.PLLSTATE

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Returns the status of the PLL |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| See Also | FBUS.PARAM1 TO FBUS.PARAM10 (→ p. 514), Fieldbus documentation |
| Start Version | M_01-02-00-000 |

Description

FBUS.PLLSTATE returns the status of the phase locked loop (PLL). The PLL states are as follows:

| PLL State | Description |
|-----------------------------|--|
| PLL not activated | This state is set using FBUS.PARAM02. Not each fieldbus or operation mode needs synchronization. |
| PLL activated, but unlocked | The PLL is activated but has not yet been locked successfully. This state is related to the fieldbus master, as well as to the fieldbus mode of operation. |
| PLL activated and locked | PLL is fully operational and locked |

19.15.3 FBUS.PLLTHRESH

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets number of successful synchronized cycles needed to lock the PLL. |
| Units | N/A |
| Range | 0 to 10,000 |
| Default Value | 0 |
| Data Type | Integer, U32 |
| See Also | Appendix B: Fieldbus Manuals |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 50B1h/0 | M_01-00-00-000 |

Description

This parameter sets number of successful synchronized cycles needed to lock the PLL.

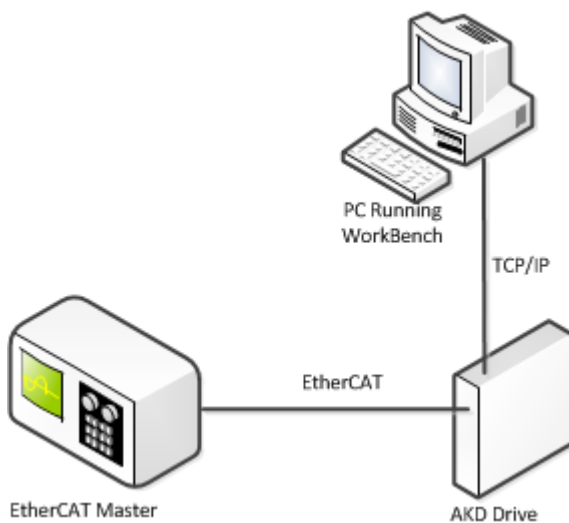
19.15.4 FBUS.PROTECTION

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Controls which parameters are blocked from being accessed through telnet while a fieldbus is operational. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-08-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3498h/0 | M_01-08-00-000 |

Description

When an AKD PDMM is a slave node on a fieldbus (for example an AKD PDMM is connected to an EtherCAT master) and EWV¹ is simultaneously connected to an AKD PDMM via a service channel, then you have several options on how you would like the AKD PDMM to respond to commands from each of the channels.



When the fieldbus is operational the AKD PDMM uses the FBUS.PROTECTION parameter to allow the user to select how commands from EWV, or any other device sending commands to the drive through a service channel, should be handled. EWV can use either of two service channels: Telnet or ASCII SDO. Both service channels are affected identically by FBUS.PROTECTION.

The following table shows the available options. If the fieldbus is not connected or not started, then parameters executed through a service channel will not be blocked.

| Value | Description |
|-------|---|
| 0 | Motion and Tuning Commands Allowed - Commands that would interfere with motion are allowed. Gain and IO configuration changes are allowed. |

¹Embedded Workbench Views

| Value | Description |
|-------|--|
| 1 | Block Motion but Allow Tuning Commands - Commands that would interfere with motion are blocked. Gain and I/O configuration changes are allowed. |

DRV.HELP will include the [Blocked] attribute to indicate whether a parameter is being blocked. For example, if FBUS.PROTECTION is 0 and the network is operational, the following help will be shown.

```
-->DRV.HELP DRV.EN
Help for: DRV.EN [ActionCommand] [Blocked]
-->
```

If you attempt to execute a blocked parameter the following error would display.

```
-->DRV.EN
Error: [0008] Command blocked.
```

All write or command functions of a parameter will be blocked by FBUS.PROTECTION. FBUS.PROTECTION will not block the read function of any parameter.

When the fieldbus is operational, FBUS.PROTECTION is always blocked, but can be edited through the fieldbus. This ensures that the protection level can only be modified by the fieldbus when it is in control.

If a parameter is blocked through telnet, it will be accessible through the fieldbus. For example, SDO's in the case of CANopen or EtherCAT.

Parameters Affected by FBUS.PROTECTION

This table shows how the blocked attribute affects the following parameters in each state of FBUS.PROTECTION. This table is true only if FBUS.STATE = 5 (operational).

| FBUS.PROTECTION | 0 - Block Nothing | 1 - Block Motion Commands |
|-----------------------------|-------------------|---------------------------|
| DRV.EN DRV.DIS | | [Blocked] |
| FBUS.PROTECTION | | [Blocked] |
| DRV.OPMODE DRV.CMDSOURCE | | [Blocked] |
| DRV.STOP DRV.CLRFAULTS | | [Blocked] |

Related Topics

FBUS.STATE

19.15.5 FBUS.REMOTE

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Changes or shows the control of the drive (fieldbus master/telnet) |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | U8 |
| Start Version | M_01-05-06-000 |

Description

DS402 (CAN/EtherCAT):

With this parameter the bit 9 (remote) of the DS402-status word can be set directly via Telnet to show the DS402-master-system that the control is removed. The master has to react on that.

Related Topics

Fieldbus Manuals

19.15.6 FBUS.SAMPLEPERIOD

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets fieldbus sample period. |
| Units | Whole multiples of MTS 250 μ s |
| Range | 4 to 128 and value must be a multiple of 4 |
| Default Value | 32 = 2 ms |
| Data Type | U8 |
| See Also | Appendix B: Fieldbus Manuals |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 60C2h/0 | M_01-00-00-000 |

Description

This parameter sets the fieldbus cycle time. It is normally written in the startup phase of the field busses via the object 60C2 subindex 1 (interpolation time units) and 2 (interpolation time index), where the index stands for a power of 10 seconds (for instance, -3 stands for milliseconds) and the units are the counts of these units. Kollmorgen recommends the following standard cycle rates, 250 μ s (4), 500 μ s (8), 1 ms (16), 2 ms (32), 4 ms (64).

19.15.7 FBUS.STATE

| General Information | |
|---------------------|---------------------------------------|
| Type | R/O |
| Description | Reads the state of the fieldbus. |
| Units | N/A |
| Range | Depends on fieldbus. See table below. |
| Default Value | Depends on fieldbus. See table below. |
| Data Type | Integer |
| Start Version | M_01-08-00-000 |

Description

FBUS.STATE returns the current state of the fieldbus. The value returned is dependent upon the fieldbus being used.

| Value | CANopen over CAN | CANopen over EtherCAT |
|-------|------------------|-----------------------|
| -1 | Not connected | Not connected |
| 0 | Init. | - |
| 1 | N/A | Init. |
| 2 | N/A | Pre-operational |
| 3 | - | Boot |
| 4 | Stopped | Safe-operational |
| 5 | Operational | - |
| 8 | - | Operational |
| 127 | Pre-operational | - |

19.15.8 FBUS.SYNACQUIREWND

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Defines the time window that is used for the starting the synchronization. |
| Units | ns |
| Range | 5 to 100,000 ns |
| Default Value | 12,500 ns |
| Data Type | Integer, U32 |
| Start Version | M_01-14-06-001 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5411h/0 | M_01-14-06-001 |

Description

Defines the time window that is used for the starting the synchronization. If the measured (FBUS.SYNCDIST (→ p. 526)) and target time (FBUS.SYNCDIST) are within this absolute window (for a certain period) the synchronization is successful and FBUS.PLLSTATE (→ p. 517) will return the locked state.

19.15.9 FBUS.SYNCACT

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads actual distance from the desired sync distance. |
| Units | ns |
| Range | 0 to 250,000 ns |
| Default Value | 0 ns |
| Data Type | Integer, U 32 |
| See Also | Appendix B: Fieldbus Manuals |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3496h/2 | M_01-00-00-000 |

Description

This parameter reads actual distance from the desired sync distance.

19.15.10 FBUS.SYNCDIST

| General Information | |
|---------------------|---------------------------------------|
| Type | NV Parameter |
| Description | Sets time target for synchronization. |
| Units | ns |
| Range | 0 to 2,500,000 ns |
| Default Value | 100,000 ns |
| Data Type | Integer, U32 |
| See Also | Appendix B: Fieldbus Manuals |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3496h/1 | M_01-00-00-000 |

Description

This parameter sets time target for synchronization.

19.15.11 FBUS.SYNCLOCKWND

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Defines the time window that is used for keeping the synchronization running. |
| Units | ns |
| Range | 5 to 50,000 ns |
| Default Value | 8,750 ns |
| Data Type | Integer, U32 |
| Start Version | M_01-14-06-001 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5411h/0 | M_01-14-06-001 |

Description

Defines the time window that is used for keeping the synchronization running. If the measured time (FBUS.SYNCDIST (→ p. 526)) and target time (FBUS.SYNCDIST) are not within this absolute window (for a certain period), the synchronization fails and the fieldbus FBUS.PLLSTATE (→ p. 517) changes to the “unlocked” state.

19.15.12 FBUS.SYNCWND

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets symmetrically arranged window around the desired sync distance. |
| Units | ns |
| Range | 0 to 1,000,000 ns |
| Default Value | 50,000 ns |
| Data Type | Integer, U2 |
| See Also | Appendix B: Fieldbus Manuals |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3496h/3 | M_01-00-00-000 |

Description

This parameter sets symmetrically arranged window around the desired sync distance.

19.15.13 FBUS.TYPE

| General Information | |
|---------------------|---------------------------------|
| Type | R/O Parameter |
| Description | Shows the active fieldbus type. |
| Units | N/A |
| Range | 0 to 6 |
| Default Value | 0 |
| Data Type | U8 |
| See Also | Fieldbus Manuals |
| Start Version | M_01-00-00-000 |

Description

FBUS.TYPE shows the active fieldbus type.

0 = Analog

1 = SynqNet

2 = EtherCAT

3 = CANopen

4 = EthernetIP

5 = Profinet

6 = sercos® III

19.16 GUI Parameters

GUI parameters are used within EWV¹ for data reporting and data storage.

| | |
|-----------------------------------|------------|
| 19.16.1 GUI.DISPLAY | 531 |
| 19.16.2 GUI.DISPLAYS | 532 |
| 19.16.3 GUI.PARAM01 | 533 |
| 19.16.4 GUI.PARAM02 | 534 |
| 19.16.5 GUI.PARAM03 | 535 |
| 19.16.6 GUI.PARAM04 | 536 |
| 19.16.7 GUI.PARAM05 | 537 |
| 19.16.8 GUI.PARAM06 | 538 |
| 19.16.9 GUI.PARAM07 | 539 |
| 19.16.10 GUI.PARAM08 | 540 |
| 19.16.11 GUI.PARAM09 | 541 |
| 19.16.12 GUI.PARAM10 | 542 |

¹Embedded Workbench Views

19.16.1 GUI.DISPLAY

| General Information | |
|---------------------|---------------------------|
| Type | R/O Parameter |
| Description | Reads drive display data. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | Display |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

This parameter reports to the GUI what the drive currently is displaying. For all GUI commands, the data is compressed and formatted for the GUI, not for the user.

19.16.2 GUI.DISPLAYS

| General Information | |
|---------------------|---------------------------|
| Type | R/O Parameter |
| Description | Reads drive display data. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | Display |
| Start Version | M_01-09-06-000 |

Description

This parameter reports to the GUI what the drive currently is displaying. For all GUI commands, the data is compressed and formatted for the GUI, not for the user.

19.16.3 GUI.PARAM01

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Used by the GUI to store data. |
| Units | N/A |
| Range | 2,147,483,648 to 2,147,483,647 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

19.16.4 GUI.PARAM02

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Used by the GUI to store data. |
| Units | N/A |
| Range | 2,147,483,648 to 2,147,483,647 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

19.16.5 GUI.PARAM03

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Used by the GUI to store data. |
| Units | N/A |
| Range | 2,147,483,648 to 2,147,483,647 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

19.16.6 GUI.PARAM04

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Used by the GUI to store data. |
| Units | N/A |
| Range | 2,147,483,648 to 2,147,483,647 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

19.16.7 GUI.PARAM05

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Used by the GUI to store data. |
| Units | N/A |
| Range | 2,147,483,648 to 2,147,483,647 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

19.16.8 GUI.PARAM06

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Used by the GUI to store data. |
| Units | N/A |
| Range | 2,147,483,648 to 2,147,483,647 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

19.16.9 GUI.PARAM07

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Used by the GUI to store data. |
| Units | N/A |
| Range | 2,147,483,648 to 2,147,483,647 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

19.16.10 GUI.PARAM08

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Used by the GUI to store data. |
| Units | N/A |
| Range | 2,147,483,648 to 2,147,483,647 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

19.16.11 GUI.PARAM09

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Used by the GUI to store data. |
| Units | N/A |
| Range | 2,147,483,648 to 2,147,483,647 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

19.16.12 GUI.PARAM10

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Used by the GUI to store data. |
| Units | N/A |
| Range | 2,147,483,648 to 2,147,483,647 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

19.17 HWLS Parameters

This section describes the HWLS parameters.

| | |
|------------------------------------|------------|
| 19.17.1 HWLS.NEGSTATE | 544 |
| 19.17.2 HWLS.POSSTATE | 545 |

19.17.1 HWLS.NEGSTATE

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the status of the negative hardware limit switch. |
| Units | 0 to 1 |
| Range | N/A |
| Default Value | Integer |
| Data Type | HWLS.POSSTATE (→ p. 545) |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

HWLS.NEGSTATE reads the status of the negative HW limit switch as follows:

0 = Low

1 = High

Related Topics

19.17.2 HWLS.POSSTATE

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the status of the positive hardware limit switch. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | N/A |
| Data Type | Integer |
| See Also | HWLS.NEGSTATE (→ p. 544) |
| Start Version | M_01-00-00-000 |

Description

HWLS.POSSTATE reads the status of the positive hardware limit switch as follows:

0 = Low

1 = High

Related Topics

19.18 IL Parameters

This section describes the IL parameters.

| | |
|----------------------------------|-----|
| 19.18.1 IL.BUSFF | 548 |
| 19.18.2 IL.CMD | 549 |
| 19.18.3 IL.CMDACC | 550 |
| 19.18.4 IL.CMDU | 551 |
| 19.18.5 IL.DCMD | 552 |
| 19.18.6 IL.DFB | 553 |
| 19.18.7 IL.DIFOLD | 554 |
| 19.18.8 IL.FB | 555 |
| 19.18.9 IL.FBSOURCE | 556 |
| 19.18.10 IL.FF | 557 |
| 19.18.11 IL.FOLDFTHRESH | 558 |
| 19.18.12 IL.FOLDFTHRESHU | 559 |
| 19.18.13 IL.FOLDWTHRESH | 560 |
| 19.18.14 IL.FRICTION | 561 |
| 19.18.15 IL.IFOLD | 562 |
| 19.18.16 IL.IUFB | 563 |
| 19.18.17 IL.IVFB | 564 |
| 19.18.18 IL.KACFF | 565 |
| 19.18.19 IL.KBUSFF | 566 |
| 19.18.20 IL.KP | 567 |
| 19.18.21 IL.KPDRATIO | 568 |
| 19.18.22 IL.KPLOOKUPINDEX | 569 |
| 19.18.23 IL.KPLOOKUPVALUE | 570 |
| 19.18.24 IL.KPLOOKUPVALUES | 571 |
| 19.18.25 IL.KPSOURCE | 572 |
| 19.18.26 IL.KVFF | 573 |
| 19.18.27 IL.LIMITN | 574 |
| 19.18.28 IL.LIMITP | 575 |
| 19.18.29 IL.MFOLDD | 576 |
| 19.18.30 IL.MFOLDR | 577 |
| 19.18.31 IL.MFOLDT | 578 |
| 19.18.32 IL.MI2T | 579 |

| | |
|--------------------------------------|------------|
| 19.18.33 IL.MI2TWTHRESH | 580 |
| 19.18.34 IL.MIFOLD | 581 |
| 19.18.35 IL.MIMODE | 582 |
| 19.18.36 IL.OFFSET | 583 |
| 19.18.37 IL.VCMD | 584 |
| 19.18.38 IL.VLIMIT | 585 |
| 19.18.39 IL.VUFB | 586 |
| 19.18.40 IL.VVFB | 587 |

19.18.1 IL.BUSFF

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Displays the current feedforward value injected by the fieldbus. |
| Units | Arms |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| See Also | IL.KBUSFF (→ p. 566) |
| Start Version | M_01-00-00-000 |

Description

This parameter displays the current feedforward value injected by the fieldbus.

Related Topics

- 1 Current Loop

19.18.2 IL.CMD

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the value of the q-component current command. |
| Units | Arms |
| Range | ± Drive peak current (DRV.IPEAK) |
| Default Value | N/A |
| Data Type | Float |
| See Also | DRV.IPEAK (→ p. 417) |
| Start Version | M_01-00-00-000 |

Description

IL.CMD displays the q-component current command value of the current loop after any limitation (such as a parameter setting or I^2t calculation).

IL.CMD is limited also by motor peak current, IL.LIMITN (→ p. 574) and IL.LIMITP (→ p. 575).

Related Topics

- 1 Current Loop

19.18.3 IL.CMDACC

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Returns the commanded acceleration from the trajectory generator. |
| Units | Depends on UNIT.ACCLINEAR or UNIT.ACCROTARY. |
| Range | +/- 64bit |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-08-03-000 |

| Fieldbus | Index/Subindex | Is 64 bit? | Attributes | Signed? |
|----------|----------------|------------|------------|---------|
| Modbus | 1760 | Yes | 64-bit | Yes |

Description

Use IL.CMDACC to monitor the commanded acceleration for a given motion. This keyword is useful for tuning acceleration feedforward (IL.KACFF).

Related Topics

IL.KACFF (→ p. 565) | Tuning Guide (→ p. 132)

19.18.4 IL.CMDU

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the user current command. |
| Units | Arms |
| Range | Minimum range value = maximum of IL.LIMITN and -MOTOR.IPEAK Maximum range value = minimum of IL.LIMITP and MOTOR.IPEAK |
| Default Value | 0 Arms |
| Data Type | Float |
| See Also | DRV.IPEAK (→ p. 417), DRV.OPMODE (→ p. 430), DRV.CMDSOURCE (→ p. 375) |
| Start Version | M_01-00-00-000 |

Description

This parameter sets the user current command value.

The current command value, which is provided to the current loop (IL.CMD), can be limited further using a parameter setting or I^2t calculation. IL.CMDU is limited also by motor peak current, IL.LIMITN (→ p. 574) and IL.LIMITP (→ p. 575).

For IPM motors (MOTOR.TYPE = 6), IL.CMDU is decomposed into d-component and q-component current values, as both are used for closed loop current control. The d-component value is used to set IL.DCMD (→ p. 552), and the q-component value is used to set IL.CMD (→ p. 549).

Related Topics

- 1 Current Loop

19.18.5 IL.DCMD

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the value of the d-component current controller inside the FPGA. |
| Units | Arms |
| Range | ± Drive peak current (DRV.IPEAK) |
| Default Value | N/A |
| Data Type | Float |
| See Also | DRV.IPEAK (→ p. 417) |
| Start Version | M_01-00-00-000 |

Description

This command displays the d-component current command value of the current loop. This parameter is commonly used with IPM motors (MOTOR.TYPE = 6), where the d-component current is controlled for closed loop current regulation.

19.18.6 IL.DFB

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the actual value of the d-component current. |
| Units | Arms |
| Range | ± Drive peak current (DRV.IPEAK) |
| Default Value | N/A |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-11-00-000 |

Description

This parameter displays the measured, de-rotated d-component current value of the motor. This parameter is commonly used with IPM motors (MOTOR.TYPE = 6), where the d-component current is controlled for closed loop current regulation.

19.18.7 IL.DIFOLD

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the drive foldback current limit. |
| Units | Arms |
| Range | 0 to 2,147,483.647 Arms |
| Default Value | N/A |
| Data Type | Float |
| See Also | Foldback |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3559h/0 | M_01-00-00-000 |

Description

IL.DIFOLD is the output of the drive foldback algorithm. It is an artificial current, which can be higher or lower than the drive peak current (DRV.IPEAK). When IL.DIFOLD is lower than the existing current limit (such as IL.LIMITP (→ p. 575)), it becomes the active current limit.

IL.DIFOLD decreases when the actual current is higher than drive continuous current and increases (up to a certain level) when the actual current is lower than drive continuous current.

Related Topics

- 1 Current Loop

19.18.8 IL.FB

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the actual value of the q-component current. |
| Units | Arms |
| Range | ± Drive peak current (DRV.IPEAK) |
| Default Value | N/A |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3558h/0 | M_01-00-00-000 |

Description

This parameter reads the measured, de-rotated actual current value of the motor.

Note: Internally the resolution of the current scale is 20130 increments. For an AKD PDMM with a peak current of 9 amps, the current resolution applied is $9/20130 = .447$ mA. For a 48 amp peak current drive, the resolution is $48/20130 = 2.38$ mA. The current scaling is hard coded and cannot be changed by decreasing the peak current settings in the drive.

Related Topics

- 1 Current Loop

19.18.9 IL.FBSOURCE

| General Information | |
|---------------------|--|
| Type | R/W |
| Description | Sets the feedback source for the current loop. Only applies when MOTOR.TYPE = 4. |
| Units | N/A |
| Range | 0 to 2 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-08-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5361h/0 | M_01-08-00-000 |

Description

This parameter selects the feedback source that current loop uses for commutation angle calculation. It applies only when MOTOR.TYPE = 4.

0 = Primary feedback

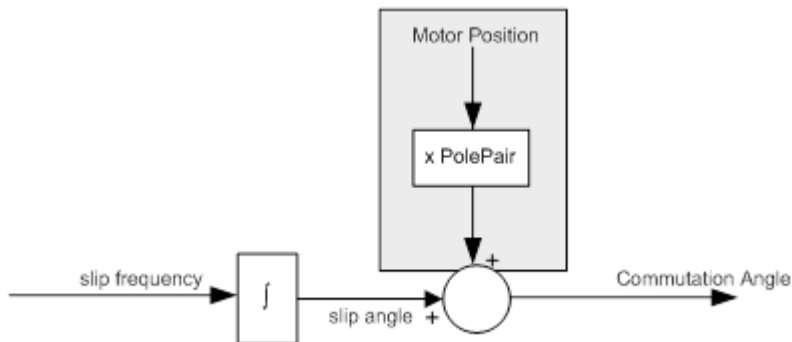
1 = Secondary feedback

2 = Tertiary feedback

When IL.FBSOURCE is changed, MOTOR.POLES and MOTOR.PHASE are updated.

| IL.FBSOURCE | MOTOR.PHASE is set to: | MOTOR.POLES is set to: |
|-------------|------------------------|------------------------|
| 0 | FB1.MOTORPHASE | FB1.MOTORPOLES |
| 1 | FB2.MOTORPHASE | FB2.MOTORPOLES |
| 2 | FB3.MOTORPHASE | FB3.MOTORPOLES |

For induction motor closed loop control the commutation angle is set as shown in the following figure.



Related Topics

MOTOR.TYPE (→ p. 629)

FB1.MOTORPOLES (→ p. 476) | FB2.MOTORPOLES (→ p. 1) | FB3.MOTORPOLES (→ p. 506)

FB1.MOTORPHASE (→ p. 475) | FB2.MOTORPHASE (→ p. 1) | FB3.MOTORPHASE (→ p. 505)

19.18.10 IL.FF

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the current loop overall feedforward value |
| Units | Arms |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| See Also | IL.KBUSFF (→ p. 566), IL.KVFF (→ p. 573), IL.OFFSET (→ p. 583), IL.FRICTION (→ p. 561), IL.KACCCFF (→ p. 565) |
| Start Version | M_01-00-00-000 |

Description

IL.FF reads the present current loop overall feed-forward value. It is the sum of the following feed-forward paths:

- Acceleration feed-forward (determined by IL.KACCCFF (→ p. 565))
- Velocity feed-forward (determined by VL.KVFF (→ p. 745))
- Friction feed-forward (determined by IL.FRICTION (→ p. 561))
- Gravity offset feed-forward (determined by IL.OFFSET (→ p. 583))
- Bus network feed-forward (determined by IL.BUSFF (→ p. 548))

IL.FF is calculated using the following variables' values at the time it is called:

$IL.FF = IL.KACCCFF * \text{Accelration Command} + VL.KVFF * \text{Velocity Command} + IL.FRICTION$ (if motion is currently commanded) $+ IL.OFFSET$ (always) $+ IL.KBUSFF$ (→ p. 566) $* IL.BUSFF$ (if fieldbus connected)

For EtherCAT:

IL.BUSFF is the injected current feedforward, which corresponds to Object 60B2h: Torque Offset (→ p. 1068). The scaling for the torque offset is done with 1/1000 rated torque. In AKD, a value of 1000 means DRV.ICONT (→ p. 414) will be commanded in IL.BUSFF (if IL.KBUSFF is set to 1).

IL.KBUSSFF is a gain setting for the feedforward value IL.BUSFF. It needs to be set to greater zero to let IL.BUSFF take effect. IL.KBUSFF can be access by SDO object 50E2h/0.

The IL.BUSFF gain can be commanded through EtherCAT in all [DS402 operating modes](#).

Related Topics

- 1 Current Loop

19.18.11 IL.FOLDFTHRESH

| General Information | |
|---------------------|---------------------------------|
| Type | R/O Parameter |
| Description | Reads the foldback fault level. |
| Units | Arms |
| Range | 0 to 500 Arms |
| Default Value | Drive peak current (DRV.IPEAK) |
| Data Type | Float |
| See Also | Foldback |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3420h/0 | M_01-00-00-000 |

Description

IL.FOLDFTHRESH is the fault level of the current foldback algorithm. If IL.IFOLD (→ p. 562) drops below the value for IL.FOLDFTHRESH, then a fault is generated and the drive is disabled.

To avoid reaching the current foldback fault level, set IL.FOLDFTHRESHU well below the continuous current value for both the drive and the motor or set the IL.FOLDFTHRESHU value to zero.

Related Topics

- 1 Current Loop

19.18.12 IL.FOLDFTHRESHU

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the user value for the foldback fault level. |
| Units | Arms |
| Range | 0 to 500 Arms |
| Default Value | Drive peak current (DRV.IPEAK) |
| Data Type | Float |
| See Also | IL.FOLDFTHRESH (→ p. 558), Foldback (→ p. 59) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3421h/0 | M_01-00-00-000 |

Description

IL.FOLDFTHRESHU is the fault level of the current foldback algorithm. The value of IL.FOLDFTHRESH is the minimum of DRV.IPEAK (→ p. 417), MOTOR.IPEAK (→ p. 607), and IL.FOLDFTHRESHU.

Related Topics

- 1 Current Loop

19.18.13 IL.FOLDWTHRESH

| General Information | |
|---------------------|----------------------------------|
| Type | NV Parameter |
| Description | Sets the foldback warning level. |
| Units | Arms |
| Range | 0 to 500 Arms |
| Default Value | 0 A |
| Data Type | Float |
| See Also | Foldback |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 355Ah/0 | M_01-00-00-000 |

Description

IL.FOLDWTHRESH is the warning level of the current foldback algorithm. When IL.IFOLD (→ p. 562) drops **below** IL.FOLDWTHRESH a warning is generated.

To ensure that the current foldback warning level is never reached, IL.FOLDWTHRESH should be set well below the continuous current value for both the drive and the motor. You can also set the IL.FOLDFTHRESH (→ p. 558) value to zero.

Related Topics

- 1 Current Loop

19.18.14 IL.FRICTION

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets friction compensation value. |
| Units | A |
| Range | 0 to the minimum of user positive current limit (IL.LIMITP) and motor peak current (MOTOR.IPEAK).IL.LIMITP (→ p. 575) |
| Default Value | 0 |
| Data Type | Float |
| See Also | IL.FF |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3422h/0 | M_01-00-00-000 |

Description

Position command derivative sign is multiplied by this value to be injected to the current command.

NOTE

IL.FRICTION is active in Position and Velocity modes (DRV.OPMODE = 1, 2), but not active in Torque mode (DRV.OPMODE = 0).

Related Topics

- 1 Current Loop

19.18.15 IL.IFOLD

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the overall foldback current limit. |
| Units | A |
| Range | 0 to 2,147,483.647 A |
| Default Value | N/A |
| Data Type | Float |
| See Also | Foldback |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3425h/0 | M_01-00-00-000 |

Description

Two current foldback algorithms run in parallel in the drive: the drive foldback algorithm and the motor foldback algorithm. Each algorithm uses different sets of parameters.

Each algorithm has its own foldback current limit, IL.DIFOLD and IL.MIFOLD. The overall foldback current limit is the minimum of the two at any given moment.

$$IL.IFOLD = \min (IL.DIFOLD, IL.MIFOLD) .$$

IL.DIFOLD is an artificial current, which can be higher or lower than the drive or motor peak current. When IL.IFOLD becomes lower than the existing current limit (such as IL.LIMITP (→ p. 575)), it becomes the active current limit.

Related Topics

- 1 Current Loop

19.18.16 IL.IUFB

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the sigma-delta measured current in the u-winding of the motor. |
| Units | A |
| Range | ± Drive peak current (DRV.IPEAK) |
| Default Value | N/A |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

This parameter displays the measured current in the u-winding of the motor.

Related Topics

- 1 Current Loop

19.18.17 IL.IVFB

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Sets the sigma-delta measured current in the u-winding of the motor. |
| Units | A |
| Range | ± Drive peak current (DRV.IPEAK) |
| Default Value | 0 A |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

IL.IVFB is an offset value that is added to the measured current in the u-winding of the motor. This value is used for compensating for an error in the current measurement. The drive measures 256 times the current in the u-winding when powering-up the drive. Afterwards, the drive calculates the average value of the measured current and uses this value for the offset value.

Related Topics

- 1 Current Loop

19.18.18 IL.KACCFF

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets current loop acceleration feedforward gain value |
| Units | mArms/(rad/s ²) |
| Range | 0 to 39113.288 mArms/(rad/s ²) |
| Default Value | 0 mArms/(rad/s ²) |
| Data Type | Float |
| See Also | IL.FF (→ p. 557) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3426h/0 | M_01-00-00-000 |

Description

This value sets the gain for the acceleration feedforward (a scaled second derivative of the position command is added to the current command value) .

This parameter is valid only in the position mode (DRV.OPMODE = 2).

Related Topics

- 1 Current Loop

19.18.19 IL.KBUSFF

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Current loops fieldbus injected feed-forward gain |
| Units | NA |
| Range | 0 to 2 |
| Default Value | 0 |
| Data Type | Float |
| See Also | IL.FF (→ p. 557), IL.BUSFF (→ p. 548) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 50E2h/0 | M_01-00-00-000 |

Description

This parameter scales the feedforward term added by the fieldbus to the current command. The nominal feed-forward value can be multiplied by this gain value.

This parameter is only used in the position mode (DRV.OPMODE = 2).

Related Topics

- 1 Current Loop

19.18.20 IL.KP

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the proportional gain of the q-component of the PI regulator. |
| Units | V/A |
| Range | 0 to 2,000 V/A |
| Default Value | Read from the motor or, if no memory, 50.009 V/A |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3598h/0 | M_01-00-00-000 |

Description

IL.KP is used to modify the proportional gain of the PI-loop that controls the q-component of the current.

Related Topics

- 1 Current Loop

19.18.21 IL.KPDRATIO

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the proportional gain of the d-component current PI-regulator as a percentage of IL.KP |
| Units | N/A |
| Range | 0 to 100 |
| Default Value | 1 |
| Data Type | Float |
| See Also | IL.KP (→ p. 567) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3596h/0 | M_01-00-00-000 |

Description

This parameter modifies the proportional gain of the PI-loop, which controls the d-component of the current.

Related Topics

- 1 Current Loop

19.18.22 IL.KPLOOKUPINDEX

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the index into the Current Loop Gain Scheduling Table. |
| Units | N/A |
| Range | 0 to 255 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | IL.KPLOOKUPVALUE (→ p. 570)IL.KPLOOKUPVALUES (→ p. 571)IL.KP (→ p. 567) |
| Start Version | M_01-04-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5265h/0 | M_01-04-00-000 |

Description

This parameter sets the index into the Current Loop Gain Scheduling Table. The table is 256 records long, spanning 0 A to $1.62 * \text{DRV.IPEAK}$.

To determine the level of current that corresponds to a table index, use the following equation:

$$\text{IL.CMD} = (\text{Table Index} / 157) * \text{DRV.IPEAK}$$

Related Topics

Current Loop

19.18.23 IL.KPLOOKUPVALUE

| General Information | |
|---------------------|---|
| Type | R/W if motor memory not supported. R/O if motor memory supported. |
| Description | Sets the value of the current loop gain scheduling index. |
| Units | % |
| Range | 0 to 100.000% |
| Default Value | 0 |
| Data Type | Float |
| See Also | IL.KPLOOKUPINDEX (→ p. 569), IL.KPLOOKUPVALUES (→ p. 571), IL.KP (→ p. 567) |
| Start Version | M_01-04-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5266h/0 | M_01-04-00-000 |

Description

This parameter sets the value at the current index into the Current Loop Gain Scheduling Table. The table is 256 records long, spanning 0 A to $1.62 * \text{DRV.IPEAK}$. The value can range from 0% to 100% and determines what percentage of IL.KP will be applied to the current loop.

If the feedback device supports motor memory ($\text{FB1.MEMVER} \geq 0.03$ and $\text{MOTOR.AUTOSSET} = 1$) this parameter is read only. Each index value is automatically calculated and set. This calculation depends on the value of MOTOR.LISAT (→ p. 611).

If the feedback device does not support motor memory, this parameter is read/write. Each index value must be entered manually. Use the following equation to determine what level of current corresponds to a table index:

$$\text{IL.CMD} = (\text{Table Index}/157) * \text{DRV.IPEAK}$$

Example

Assume:

$\text{DRV.IPEAK} = 9 \text{ A}$

$\text{IL.KPLOOKUPINDEX} = 100$

$\text{IL.KPLOOKUPVALUE} = 50$

$\text{IL.KP} = 240$

When $\text{IL.CMD} = 100/157 * 9 = 5.73 \text{ A}$, IL.KP will not be 240, but will be $50\% * 240 = 120$.

Related Topics

Current Loop

19.18.24 IL.KPLOOKUPVALUES

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Gets the Current Loop Gain Scheduling Table. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | Table |
| See Also | IL.KPLOOKUPINDEX (→ p. 569), IL.KPLOOKUPVALUE (→ p. 570), IL.KP (→ p. 567) |
| Start Version | M_01-04-00-000 |

Description

Retrieves the Current Loop Gain Scheduling Table in a comma delimited table.

This table is 256 records long, and the table will return values in the following format:

```
-->IL.KPLOOKUPVALUES
```

```
Index Value
```

```
0, 100.000
```

```
1, 100.000
```

```
2, 100.000
```

```
3, 100.000
```

```
4, 100.000
```

```
5, 100.000
```

```
6, 100.000
```

```
7, 100.000
```

```
8, 100.000
```

```
9, 100.000
```

```
10, 100.000
```

Related Topics

Current Loop

19.18.25 IL.KPSOURCE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Selects a source for the IL.KP value when MOTOR.AUTOSSET is enabled |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-14-06-000 |

Description

Sets the source of the IL.KP (→ p. 567) value when MOTOR.AUTOSSET (→ p. 592) is enabled.

| | |
|---|---|
| 0 | The Autoset feature calculates IL.KP from motor parameters |
| 1 | IL.KP is read from NV (or set to the default value if no value is stored) |

19.18.26 IL.KVFF

| General Information | |
|---------------------|--|
| Type | R/W |
| Description | Current loop velocity feed-forward gain. |
| Units | N/A |
| Range | 0 to 2 |
| Default Value | 0 |
| Data Type | Float |
| See Also | IL.FF (→ p. 557) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3544h/0 | M_01-00-00-000 |

Description

This parameter sets the gain for the velocity loop feedforward. The nominal feedforward value can be multiplied by this gain value.

This parameter is only used in position mode (DRV.OPMODE (→ p. 430) = 2).

Related Topics

- 1 Current Loop

19.18.27 IL.LIMITN

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the negative user (application-specific) current limit. |
| Units | A |
| Range | Negative drive peak current (DRV.IPEAK) to 0 A |
| Default Value | Negative drive peak current (DRV.IPEAK) |
| Data Type | Float |
| See Also | IL.LIMITP (→ p. 575) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 356Fh/0 | M_01-00-00-000 |

Description

This parameter sets the negative user limit clamp value of the torque-producing q-component current command (IL.CMD (→ p. 549)). The current command is additionally limited by the motor peak current setting (MOTOR.IPEAK (→ p. 607)) and by the present value of the foldback I²t peak motor current protection.

NOTE

This current limit is overwritten by MT.FIXEDSTOPCURRENT while a fixed stop motion task is running in a negative direction. Once the motion task is complete, the limit is restored to the previous IL.LIMITN value.

Related Topics

- 1 Current Loop

19.18.28 IL.LIMITP

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the positive user (application-specific) current limit. |
| Units | A |
| Range | 0 A to drive peak current (DRV.IPEAK) |
| Default Value | Drive peak current (DRV.IPEAK) |
| Data Type | Float |
| See Also | IL.LIMITN (→ p. 574) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 356Eh/0 | M_01-00-00-000 |

Description

This parameter sets the positive user limit clamp value of the torque-producing q-component current command (IL.CMD (→ p. 549)). The current command is additionally limited by the motor peak current setting (MOTOR.IPEAK (→ p. 607)) and by the present value of the foldback I²t peak motor current protection.

NOTE

This current limit is overwritten by MT.FIXEDSTOPCURRENT while a fixed stop motion task is running in a position direction. Once the motion task is complete, the limit is restored to the previous IL.LIMITP value.

Related Topics

- 1 Current Loop

19.18.29 IL.MFOLDD

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Sets the motor foldback maximum time at motor peak current. |
| Units | s |
| Range | 0.1 to 2400 s |
| Default Value | 10 s |
| Data Type | Float |
| See Also | Foldback |
| Start Version | M_01-00-00-000 |

Description

IL.MFOLDD sets the maximum time allowed for the motor to remain at peak current before starting to fold towards the motor continuous current. When at motor peak current, IL.MFOLDD is the amount of time before the foldback algorithm starts to reduce the current.

Related Topics

- 1 Current Loop

19.18.30 IL.MFOLDR

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Sets the motor foldback recovery time. |
| Units | s |
| Range | 0.1 to 65,535 s |
| Default Value | Calculated from other foldback parameters. |
| Data Type | Float |
| See Also | Foldback |
| Start Version | M_01-00-00-000 |

Description

IL.MFOLDR sets the recovery time for the motor foldback algorithm. If 0 current is applied for at least the recovery time duration, it is possible to apply motor peak current for the duration of IL.MFOLDD time.

The IL.MFOLDR value is automatically calculated from other foldback parameters.

Related Topics

- 1 Current Loop

19.18.31 IL.MFOLDT

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Sets the motor foldback time constant of the exponential current drop (foldback). |
| Units | s |
| Range | 0.1 to 2,400 s |
| Default Value | 10 s |
| Data Type | Float |
| See Also | Foldback |
| Start Version | M_01-00-00-000 |

Description

IL.MFOLDT sets the time constant of the exponential drop (foldback) of the current towards motor continuous current.

Related Topics

- 1 Current Loop

19.18.32 IL.MI2T

| General Information | |
|---------------------|--------------------------------------|
| Type | R/O parameter |
| Description | Motor I2t load. |
| Units | % |
| Range | 0 to 100% |
| Default Value | N/A |
| Data Type | Float |
| See Also | IL.MIMODE, IL.MI2TWTHRESH (→ p. 580) |
| Start Version | M_01-04-01-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3427h/3 | M_01-00-00-000 |

Description

This parameter returns the motor I2t load in percent. The supplied current will be limited by IL.MIFOLD to MOTOR.ICONT case that the load reaches a value of 100%. The current limit IL.MIFOLD will be restored to MOTOR.IPEAK in case that the load falls under 95%.

19.18.33 IL.MI2TWTRESH

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Motor I2t load warning threshold. |
| Units | % |
| Range | 0 to 100% |
| Default Value | N/A |
| Data Type | Integer |
| See Also | IL.MIMODE, IL.MI2T (→ p. 579) |
| Start Version | M_01-04-01-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3427h/2 | M_01-00-00-000 |

Description

This parameter defines a warning threshold for the IL.MI2T value. A warning n309 will be generated as soon as the IL.MI2T exceeds the IL.MI2TWTRESH value. The warning n309 will be cleared as soon as IL.MI2T falls below the threshold.

Related Topics

Foldback (→ p. 59)

19.18.34 IL.MIFOLD

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Sets the motor foldback current limit. |
| Units | A |
| Range | 0 to 2147483.647 A |
| Default Value | N/A |
| Data Type | Float |
| See Also | Foldback |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 35A4h/0 | M_01-00-00-000 |

Description

IL.MIFOLD sets the output of the motor foldback algorithm. It is an artificial current, which can be higher or lower than the motor peak current. When IL.MIFOLD becomes lower than the existing current limit (IL.LIMITP (→ p. 575)) it becomes the active current limit.

IL.MIFOLD decreases when the actual current is higher than motor continuous current and increases (up to a certain level) when the actual current is lower than the motor continuous current.

Related Topics

- 1 Current Loop

19.18.35 IL.MIMODE

| General Information | |
|---------------------|---|
| Type | NV parameter |
| Description | Motor protection mode. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | IL.MI2T (→ p. 579), IL.MI2TWTHRESH (→ p. 580) |
| Start Version | M_01-04-01-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3427h/1 | M_01-00-00-000 |

Description

This parameter determines the method of the motor protection.

0 – The motor foldback mechanism is responsible for protecting the motor from overload.

1 – The Motor I2t algorithm is responsible for protecting the motor from overload.

Related Topics

Foldback (→ p. 59)

19.18.36 IL.OFFSET

| General Information | |
|---------------------|---|
| Type | RW Parameter |
| Description | A constant current command added to compensate for gravity. |
| Units | A |
| Range | [IL.LIMITN (→ p. 574) to IL.LIMITP (→ p. 575)] |
| Default Value | 0 A |
| Data Type | Float |
| See Also | IL.FF |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3423h/0 | M_01-00-00-000 |

Description

This value is added to the overall current loop feedforward value.

Related Topics

- 1 Current Loop

19.18.37 IL.VCMD

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the output of the q-component PI regulator. |
| Units | Vrms |
| Range | 0 Vrms to bus voltage |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

Description

Reads the output of the current loop that controls the q-component of the current.

Related Topics

- 1 Current Loop

19.18.38 IL.VLIMIT

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the velocity limit for opmode 0 (torque). |
| Units | Depends on UNIT.VROTARY or UNIT.VLINEAR Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: counts/s, mm/s, μ m/s |
| Range | Rotary: 0.000 to 15,000.000 rpm 0.000 to 250.000 rps 0.000 to 90,000.000 deg/s 0.000 to 1,250.000 custom units/s 0.000 to 1570.796 rad/s Linear: 0.000 to 1,073,741,824.000 counts/s 0.000 to 250.000*MOTOR.PITCH (\rightarrow p. 617) mm/sec 0.000 to 250,000.000*MOTOR.PITCH μ m/s |
| Default Value | Rotary: 0.000 rpm 0.000 rps 0.000 deg/s 0.000 custom units/s 0.000 rad/s Linear: 0.000 counts/s 0.000*MOTOR.PITCH (\rightarrow p. 617) mm/sec 0.000 *MOTOR.PITCH μ m/sec |
| Data Type | Float |
| Start Version | M_01-14-04-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5408h/0 | M_01-14-04-000 |

Description

IL.VLIMIT sets the velocity limit for operation in torque mode.

When set to a non-zero value, the current supplied to the motor will be limited to prevent the drive from accelerating beyond $-IL.VLIMIT / IL.VLIMIT$.

Related Topics

- 1 Current Loop

19.18.39 IL.VUFB

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the measured voltage on the u-winding of the motor. |
| Units | V |
| Range | -1200*VBusScale to +1200*VBusScale |
| Default Value | N/A |
| Data Type | Integer |
| See Also | IL.VVFB (→ p. 587) |
| Start Version | M_01-00-00-000 |

Description

Reads the measured voltage on the u-winding of the motor.

Related Topics

- 1 Current Loop

19.18.40 IL.VVFB

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the measured voltage on the v-winding of the motor. |
| Units | V |
| Range | -1200*VBusScale to +1200*VBusScale |
| Default Value | N/A |
| Data Type | Integer |
| See Also | IL.VUFB (→ p. 586) |
| Start Version | M_01-00-00-000 |

Description

The range for this parameter depends on whether the drive model is an MV/240 VAC or an HV/480 VAC.

The VBusScale parameter sets the drive model:

MV/240 VAC: VBusScale = 1

HV/480 VAC: VBusScale = 2

VBusScale is used for multiple parameter ranges that are model dependent, such as IL.KP.

Related Topics

- 1 Current Loop

19.19 LOAD Parameters

This section describes the LOAD parameters.

| | |
|-----------------------------------|------------|
| 19.19.1 LOAD.INERTIA | 589 |
|-----------------------------------|------------|

19.19.1 LOAD.INERTIA

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the load inertia. |
| Units | kgcm ² for rotary motors kg for linear motors |
| Range | 1 to 1,000,000 kgcm ² or kg |
| Default Value | 0 kgcm ² or kg |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-03-06-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 525Fh/0 | M_01-03-06-000 |

Description

LOAD.INERTIA sets the load inertia.

Related Topics

Motor (→ p. 50)

19.20 MOTOR Parameters

This section describes the MOTOR parameters.

| | |
|--------------------------------|-----|
| 19.20.1 MOTOR.AUSET | 592 |
| 19.20.2 MOTOR.BRAKE | 594 |
| 19.20.3 MOTOR.BRAKEIMM | 595 |
| 19.20.4 MOTOR.BRAKERLS | 596 |
| 19.20.5 MOTOR.BRAKESTATE | 597 |
| 19.20.6 MOTOR.CTF0 | 598 |
| 19.20.7 MOTOR.FIELDWEAKENING | 599 |
| 19.20.8 MOTOR.ICONT | 600 |
| 19.20.9 MOTOR.IDDATAVALID | 601 |
| 19.20.10 MOTOR.IDMAX | 602 |
| 19.20.11 MOTOR.IMID | 603 |
| 19.20.12 MOTOR.IMTR | 604 |
| 19.20.13 MOTOR.INERTIA | 606 |
| 19.20.14 MOTOR.IPEAK | 607 |
| 19.20.15 MOTOR.KE | 608 |
| 19.20.16 MOTOR.KT | 609 |
| 19.20.17 MOTOR.LDLL | 610 |
| 19.20.18 MOTOR.LISAT | 611 |
| 19.20.19 MOTOR.LQLL | 612 |
| 19.20.20 MOTOR.NAME | 613 |
| 19.20.21 MOTOR.PHASE | 614 |
| 19.20.22 MOTOR.PHSADV1 | 615 |
| 19.20.23 MOTOR.PHSADV2 | 616 |
| 19.20.24 MOTOR.PITCH | 617 |
| 19.20.25 MOTOR.POLES | 618 |
| 19.20.26 MOTOR.R | 619 |
| 19.20.27 MOTOR.RTYPE | 620 |
| 19.20.28 MOTOR.SUPPORTEDPARAMS | 621 |
| 19.20.29 MOTOR.TBRAKEAPP | 622 |
| 19.20.30 MOTOR.TBRAKERLS | 623 |
| 19.20.31 MOTOR.TBRAKETO | 624 |
| 19.20.32 MOTOR.TEMP | 625 |

| | |
|---------------------------------------|------------|
| 19.20.33 MOTOR.TEMPC | 626 |
| 19.20.34 MOTOR.TEMPFAULT | 627 |
| 19.20.35 MOTOR.TEMPWARN | 628 |
| 19.20.36 MOTOR.TYPE | 629 |
| 19.20.37 MOTOR.VMAX | 631 |
| 19.20.38 MOTOR.VOLTMAX | 632 |
| 19.20.39 MOTOR.VOLTMIN | 633 |
| 19.20.40 MOTOR.VOLTRATED | 634 |
| 19.20.41 MOTOR.VRATED | 635 |

19.20.1 MOTOR.AUTOSET

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Determines which drive parameters are calculated automatically. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Boolean |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3451h/0 | M_01-00-00-000 |

Description

This parameter determines whether or not certain drive parameters (see lists below) are calculated automatically. When MOTOR.AUTOSET = 1 these parameters are automatically calculated from the motor ID data (read from memory-supporting feedback devices). Parameters loaded from motor memory are then read-only and cannot be modified as long as MOTOR.AUTOSET = 1. If MOTOR.AUTOSET = 0, parameters are not loaded from motor memory and must be configured by the user. Parameters which are not loaded from motor memory are always read-write.

⚠ WARNING

When loading a parameter file to a new device, if MOTOR.AUTOSET = 0 then motor parameters should be checked to make sure they are properly configured for the motor on the new device.

Three conditions determine which parameters are calculated automatically by MOTOR.AUTOSET = 1. See the lists below to determine which parameters will automatically be configured from the motor ID data.

The following parameters may be configured automatically when MOTOR.AUTOSET = 1:

- MOTOR.BRAKE (→ p. 594)
- MOTOR.CTF0 (→ p. 598)
- MOTOR.ICONT (→ p. 600)
- MOTOR.INERTIA (→ p. 606)
- MOTOR.IPEAK (→ p. 607)
- MOTOR.KE (→ p. 608)
- MOTOR.KT (→ p. 609)
- MOTOR.LQLL (→ p. 612)
- MOTOR.NAME (→ p. 613)
- MOTOR.POLES (→ p. 618)
- MOTOR.R (→ p. 619)

If the motor's FB1.MEMVER > 0.01, AKD PDMM will also configure:

- MOTOR.PHASE (→ p. 614)
- MOTOR.TBRAKEAPP (→ p. 622)
- MOTOR.PHASE (→ p. 614)
- MOTOR.VRATED (→ p. 635)
- MOTOR.VOLTRATED (→ p. 634)

For firmware versions 01-09-00-000 and later, AKD PDMM will also configure:

- MOTOR.RTYPE (→ p. 620)
- MOTOR.TEMPFAULT (→ p. 627)

Related Topics

7.1 Motor

19.20.2 MOTOR.BRAKE

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the presence or absence of a motor brake. Can also be set to bypass F305. |
| Units | N/A |
| Range | 0, 1, 100 |
| Default Value | 0 |
| Data Type | Boolean |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3587h/0 | M_01-00-00-000 |

Description

The MOTOR.BRAKE parameter notifies the firmware whether a brake exists or not. It does not apply or release the brake. If a brake is found to be present, the firmware considers hardware indications regarding the brake circuits (such as open circuit or short circuit). If a brake does not exist, then the firmware ignores the hardware indications since they are irrelevant.

| Value | Status |
|-------|--|
| 0 | Motor brake does not exist. |
| 1 | Motor brake exists and brake hardware circuitry checks are enabled. |
| 100 | For special, low current, brake applications, the F305 (→ p. 184) fault can be bypassed with this setting. |

Enabling the MOTOR.BRAKE (value set to 1) when no motor brake exists creates a fault.

The motor brake is polled every 16 ms.

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

7.1 Motor

19.20.3 MOTOR.BRAKEIMM

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Brake Immediately: in the case of a drive disable, apply the brake as soon as the power stage disables. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 (Inactive) |
| Data Type | Boolean |
| See Also | N/A |
| Start Version | M_01-05-11-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5268h/0 | M_01-05-11-000 |

Description

Sets the brake to apply as soon as the power stage disables. The brake will not apply immediately when the disable command is issued, but at the point when the power stage actually disables. The power stage can disable due to user command (DRV.DIS), digital input, fieldbus, STO, or fault. The point at which the power stage disables depends on the disable mode settings or type of fault for each case.

| Value | Description |
|-------|--|
| 0 | Inactive |
| 1 | Brake as soon as power stage disables. |

Related Topics

Motor (→ p. 50) | DRV.DISMODE (→ p. 385) | CS.VTHRESH (→ p. 332) | CS.TO (→ p. 331) | MOTOR.TBRAKETO (→ p. 624) | DRV.DISTO (→ p. 388)

19.20.4 MOTOR.BRAKERLS

| General Information | |
|---------------------|--|
| Type | Command |
| Description | Allows a user to release or apply the motor brake. |
| Units | N/A |
| Range | 0 to 2 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3450h/0 | M_01-07-04-000 |

Description

This command allows a user to release or apply the motor brake.

0 = Drive controls the brake.

1 = Brake is released.

2 = Brake is applied.

NOTE

If a vertical axis is stationary and disabled, toggling the brake with MOTOR.BRAKERLS (→ p. 596) will not reapply the brake until controlled stop conditions are met (the axis actual velocity has been less than CS.VTHRESH (→ p. 332) for CS.TO (→ p. 331) milliseconds). In order for the brake to reapply immediately when commanding MOTOR.BRAKERLS, set CS.VTHRESH to a value greater than the velocity possible by the free falling axis and set CS.TO to 1 millisecond (its minimum value) prior to making the command.

NOTE

A digital input mode can also be used for the same purpose. The two mechanisms are independent.

Related Topics

Motor (→ p. 50)

19.20.5 MOTOR.BRAKESTATE

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the actual status of the motor brake. |
| Units | N/A |
| Range | Brake released or not present. Brake applied. |
| Default Value | Brake applied or not present. |
| Data Type | String |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

This parameter reads the actual status of the motor brake and can only show two states:

1 = Brake released or not present

2 = Brake applied

Related Topics

Motor (→ p. 50)

19.20.6 MOTOR.CTF0

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the thermal constant of the motor coil. |
| Units | mHz |
| Range | 0.265 to 16,000 mHz |
| Default Value | 10 mHz |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3454h/0 | M_01-00-00-000 |

Description

This parameter is used to configure the thermal constant of the motor coil, which is the break frequency of a single-pole low-pass filter model of the thermal dynamics of the motor coil.

This parameter, together with MOTOR.IPEAK (→ p. 607) and MOTOR.ICONT (→ p. 600), determine the motor foldback parameters IL.MFOLDD (→ p. 576), IL.MFOLDT (→ p. 578), and IL.MFOLDR (→ p. 577).

Calculating MOTOR.CTF0

Given a motor coil/winding thermal time constant T in seconds, then:

$$\text{MOTOR.CTF0} = 1 / (2\pi T)$$

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

7.1 Motor

19.20.7 MOTOR.FIELDWEAKENING

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Allows a surface permanent magnet motor to operate as an interior permanent magnet motor. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-13-02-000 |

Description

Allows a surface permanent magnet motor (MOTOR.TYPE = 0) to operate as an interior permanent magnet motor (MOTOR.TYPE = 6).

This keyword can only be set when MOTOR.TYPE = 0. Any other MOTOR.TYPE will result in an error and MOTOR.FIELDWEAKENING will not be applied.

Related Topics

7.1 Motor

MOTOR.TYPE (→ p. 629)

19.20.8 MOTOR.ICONT

| General Information | |
|---------------------|------------------------------------|
| Type | NV Parameter |
| Description | Sets the motor continuous current. |
| Units | A |
| Range | 0.1 to 500 A |
| Default Value | 1.0 A |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 358Eh/0 | M_01-00-00-000 |

Description

This parameter is used to configure the motor continuous current.

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

7.1 Motor

19.20.9 MOTOR.IDDATAVALID

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reports the status of the motor memory. |
| Units | N/A |
| Range | N/A |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-02-00-000 |

Description

MOTOR.IDDATAVALID reports the status of the motor memory status.

The valid values for this keyword are the following:

| Value | Description |
|-------|---|
| 0 | Error in identification |
| 1 | Success in identification |
| 2 | Identification in process |
| 3 | Identification not started yet |
| 4 | Success recognizing feedback, but failed to verify OEM data integrity |

Related Topics

Motor (→ p. 50)

19.20.10 MOTOR.IDMAX

| General Information | |
|---------------------|------------------------------|
| Type | NV Parameter |
| Description | Maximum direct axis current. |
| Units | Arms |
| Range | 0.2 to 1000 |
| Default Value | 0.5 * MOTOR.IPEAK |
| Data Type | Float |
| Start Version | M_01-11-00-000 |

Description

This parameter is supported for IPM motors (MOTOR.TYPE = 6). MOTOR.IDMAX is the maximum motor winding current in the direct (non-torque producing) axis.

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET = 1.

Related Topics

7.1 Motor | MOTOR.IPEAK (→ p. 607)

19.20.11 MOTOR.IMID

| General Information | |
|---------------------|---|
| Type | R/W |
| Description | The direct-axis current set point used for induction machine closed-loop control. |
| Units | Arms |
| Range | 0 to DRIVE.IPEAK |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-08-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5362h/0 | M_01-08-00-000 |

Description

The value of IL.IMID can be estimated using the motor name plate information.

The following rule should generally be used:

$$I_{d,rms} \cong I_R * \sqrt{1 - (\cos \varphi)^2} * 0.8$$

In which I_d is the value for IL.IMID, the preset rotor flux building current, I_r is the name plate phase current, $\cos \varphi$ is the name plate power factor at rated and operation, and **0.8** is an empirical factor (accounts for the angle error due to voltage drop on the leakage induction).

Related Topics

Motor (→ p. 50)

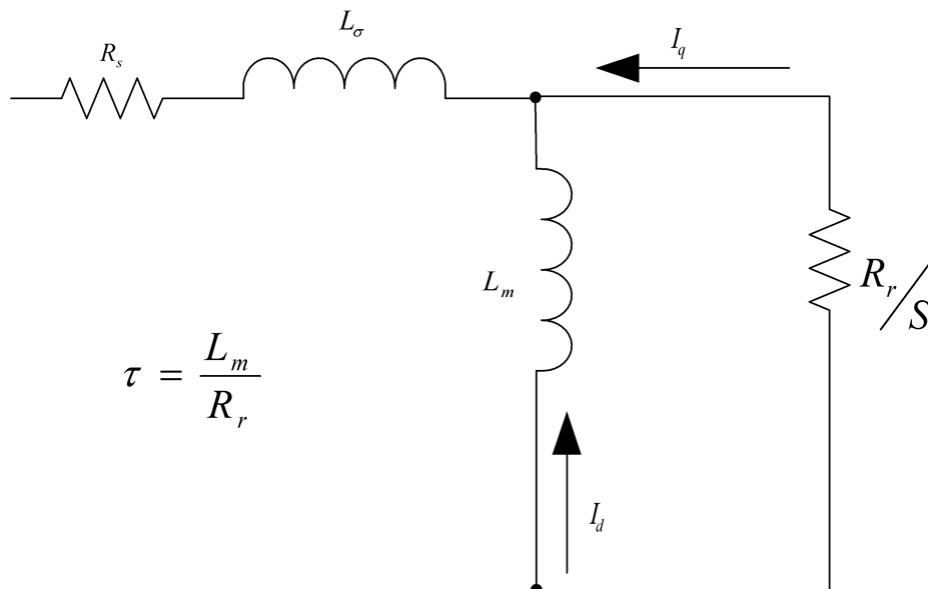
19.20.12 MOTOR.IMTR

| General Information | |
|---------------------|----------------------|
| Type | R/W |
| Description | Rotor time constant. |
| Units | Ms |
| Range | 1 to 16,000 |
| Default Value | 100 |
| Data Type | Integer |
| Start Version | M_01-08-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5360h/0 | M_01-08-00-000 |

Description

The rotor time constant is calculated as the ratio between rotor magnetizing inductance L_m and rotor resistance R_r from the following equivalent circuit of an induction machine.



Alternatively, the rotor time constant can be estimated from the rated current, magnetizing current (MOTOR.IMID (→ p. 603)) and rated slip frequency as follows:

$$\tau = \frac{\sqrt{I_{s,rms}^2 - I_{d,rms}^2}}{I_{d,rms} * \omega_{slip}}$$

where $I_{s,rms}$ is the rated current, $I_{d,rms}$ is the rated magnetizing current, and ω_{slip} is the rated slip frequency in rad/s.

NOTE

τ is in seconds for the above formulas. It needs to be converted to ms for the MOTOR.IMTR setting:
 $\text{MOTOR.IMTR(ms)} = \tau * 1000$

Related Topics

Motor (→ p. 50)

19.20.13 MOTOR.INERTIA

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the motor inertia. |
| Units | kgcm ² for rotary motors kg for linear motors |
| Range | 1 to 200,000 kgcm ² or kg |
| Default Value | 100 kgcm ² or kg |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 35ABh/0 | M_01-00-00-000 |

Description

This parameter sets the motor inertia.

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

7.1 Motor

19.20.14 MOTOR.IPEAK

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the motor peak current. |
| Units | mA |
| Range | 0.200 to 1,000 A |
| Default Value | 2.000 A |
| Data Type | Float |
| See Also | IL.LIMITP (→ p. 575), IL.LIMITN (→ p. 574) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 358Fh/0 | M_01-00-00-000 |

Description

This parameter configures the drive for the motor's peak, instantaneous-rated current. MOTOR.IPEAK is used to limit clamp the magnitude of the torque producing q-component current command (IL.CMD (→ p. 549)).

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

7.1 Motor

19.20.15 MOTOR.KE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the motor back EMF constant. |
| Units | Vpeak/krpm for Rotary Motors Vpeak/m/s for Linear Motors |
| Range | 0.0 to 100,000 |
| Default Value | 0 |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-03-06-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5260h/0 | M_01-03-06-000 |

Description

MOTOR.KE defines the back EMF constant for the motor. The back EMF constant defines how much voltage is generated at the motors coils. The relationship between MOTOR.KE and speed is described by the following equation:

$$\text{Coil Voltage} = \text{MOTOR.KE} * \text{VL.FB}$$

VL.FB is in units of krpm for rotary motors and in units of m/s for linear motors.

The maximum speed for interior permanent magnet motors (IPM) is determined by MOTOR.KE and VBUS.VALUE. An overspeed fault is generated if MOTOR.TYPE is set to 6 (IPM) and VL.FB * MOTOR.KE exceeds VBUS.VALUE.

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

Motor (→ p. 50) | VL.FB (→ p. 732) | VBUS.VALUE (→ p. 718)

19.20.16 MOTOR.KT

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the torque constant of the motor. |
| Units | Nm/A |
| Range | 0.001 Nm/A to 1,000,000.000 Nm/A for rotary motors. 0.001 Nm/A to 1,000,000.000 N/A for linear motors. |
| Default Value | 0.1 Nm/A |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3593h/0 | M_01-00-00-000 |

Description

This parameter is the torque constant of the motor in Nm/A. The value can be online checked according to the following equation:

$$K_t = 60 \cdot \sqrt{3} \cdot U_i / (2 \cdot \pi \cdot n)$$

Where:

U_i = induced voltage of the motor
 n = actual rotor velocity

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

7.1 Motor

19.20.17 MOTOR.LDLL

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Motor line to line direct axis inductance. |
| Units | mH |
| Range | 1 to 2 ³¹ |
| Default Value | 17.000 |
| Data Type | Float |
| Start Version | M_01-11-00-000 |

Description

This parameter is the motor line to line direct axis inductance. It is supported for the IPM motor type (MOTOR.TYPE = 6).

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET = 1.

Related Topics

7.1 Motor | MOTOR.LQLL (→ p. 612) | IL.KPDRATIO (→ p. 568)

19.20.18 MOTOR.LISAT

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Motor quadrature axis inductance saturation characteristic. |
| Units | Arms |
| Range | 0.2A to 2*DRV.IPEAK |
| Default Value | 1000 |
| Data Type | Float |
| Start Version | M_01-11-00-000 |

Description

This parameter is the quadrature axis inductance (L_{qll}) saturation characteristic as a function of quadrature current (I_q). It is used for q-axis current loop gain scheduling to maintain bandwidth and stability during q-axis inductance saturation. This value can also be used for q-axis flux estimation during field weakening, especially for IPM motors (MOTOR.TYPE = 6).

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET = 1.

Related Topics

7.1 Motor | MOTOR.LQLL (→ p. 612)

19.20.19 MOTOR.LQLL

| General Information | |
|---------------------|---------------------------------|
| Type | NV Parameter |
| Description | Sets the line-to-line motor Lq. |
| Units | mH |
| Range | 1 to 2 ³² mH |
| Default Value | 17.000 mH |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3455h/0 | M_01-00-00-000 |

Description

This parameter is used to configure the motor line-to-line inductance.

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

7.1 Motor

19.20.20 MOTOR.NAME

| General Information | |
|---------------------|----------------------|
| Type | NV Parameter |
| Description | Sets the motor name. |
| Units | N/A |
| Range | 11 chars |
| Default Value | N/A |
| Data Type | String |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

This parameter is used to set the motor name.

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

Motor (→ p. 50)

19.20.21 MOTOR.PHASE

| General Information | |
|---------------------|-----------------------|
| Type | NV Parameter |
| Description | Sets the motor phase. |
| Units | Electrical degrees |
| Range | 0 to 360° |
| Default Value | 0° |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 359Ch/0 | M_01-00-00-000 |

Description

This parameter sets the motor phase.

NOTE

If MOTOR.AUTOSET = 1 and FB1.MEMVER > 0.01, this parameter will be automatically configured for Kollmorgen motors.

⚠ WARNING

When loading a parameter file to a new device, if MOTOR.AUTOSET = 0 then motor parameters should be checked to make sure they are properly configured for the motor on the new device.

Related Topics

7.1 Motor

19.20.22 MOTOR.PHSADV K1

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | First order phase advance polynomial coefficient. |
| Units | N/A |
| Range | -1.0 to 1.0 |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-11-00-000 |

Description

This parameter is supported for IPM motors (MOTOR.TYPE = 6). MOTOR.PHSADV K1 is the first order polynomial coefficient used to determine the direct axis current command for commutation phase advance, according to the equation:

$$IdCmd(I_p) = PHSADV K2 * I_p^2 + PHSADV K1 * I_p$$

where I_p is the magnitude of the motor phase current:

$$I_p = \sqrt{Id^2 + Iq^2}$$

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOS E T=1.

Related Topics

7.1 Motor | MOTOR.PHSADV K2 (→ p. 616)

19.20.23 MOTOR.PHSADV2

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Second order phase advance polynomial coefficient. |
| Units | N/A |
| Range | -1.0 to 1.0 |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-11-00-000 |

Description

This parameter is supported for IPM motors (MOTOR.TYPE = 6). MOTOR.PHSADV2 is the second order polynomial coefficient used to determine the direct axis current command for commutation phase advance, according to the equation:

$$IdCmd(I_p) = PHSADV2 * I_p^2 + PHSADV1 * I_p$$

where I_p is the magnitude of the motor phase current:

$$I_p = \sqrt{Id^2 + Iq^2}$$

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

7.1 Motor | MOTOR.PHSADV1 (→ p. 615)

19.20.24 MOTOR.PITCH

| General Information | |
|---------------------|-----------------------|
| Type | NV Parameter |
| Description | Sets the motor pitch. |
| Units | µm |
| Range | 1,000 to 1,000,000 µm |
| Default Value | 1.000 µm |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 50FBh/0 | M_01-00-00-000 |

Description

This parameter defines the pole-to-pair pitch for the linear motor in micrometers.

Related Topics

Motor (→ p. 50)

19.20.25 MOTOR.POLES

| General Information | |
|---------------------|---------------------------------|
| Type | NV Parameter |
| Description | Sets the number of motor poles. |
| Units | N/A |
| Range | 0 to 510 |
| Default Value | 6 |
| Data Type | Integer |
| See Also | FB1.POLES (→ p. 486) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 359Dh/0 | M_01-00-00-000 |

Description

MOTOR.POLES sets the number of motor poles. This command is used for commutation control and represents the number of individual magnetic poles of the motor (not pole pairs). A motor's magnetic poles range from 0 to 510, which corresponds to 0 to 255 magnetic pole pairs. The division value of motor poles (MOTOR.POLES) and feedback poles (FB1.POLES) must be an integer when setting drive to enable, otherwise a fault is issued.

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

7.1 Motor

19.20.26 MOTOR.R

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the stator winding resistance phase-phase in ohms. |
| Units | Ω |
| Range | 0.001 to 650 Ω |
| Default Value | 10 Ω |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3456h/0 | M_01-00-00-000 |

Description

MOTOR.R sets the stator winding resistance phase-to-phase in ohms.

NOTE

This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET=1.

Related Topics

7.1 Motor

19.20.27 MOTOR.RTYPE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Defines the type of thermal resistor inside the motor. |
| Units | N/A |
| Range | 0 to 1 For firmware version 01-10-02-000 and later, range is 0 to 255. |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 50FEh/0 | M_01-00-00-000 |

Description

This parameter defines the type of thermal resistor used inside of the motor to measure motor temperature.

For firmware versions prior to 01-10-02-000, parameter range is 0 to 1.

0 = PTC

1 = NTC

For firmware versions 01-10-02-000 and later, parameter range is 0 to 255.

0 = Single PTC Thermistor

1 = Single NTC Thermistor

2 = KTY83-110 Thermistor

3 = KTY84-130 Thermistor

4 = PTC + KTY83-110 Thermistors

5 = Thermal switch

255 = No thermal sensor in the motor

NOTE

For firmware versions 01-09-00-000 and later, this parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET = 1.

Related Topics

Motor (→ p. 50)

19.20.28 MOTOR.SUPPORTEDPARAMS

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Lists MOTOR parameters supported by configured motor type. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| Start Version | M_01-11-00-000 |

Description

This parameter lists all the MOTOR parameters that are supported for the motor type selected by the MOTOR.TYPE parameter.

Example

The following output is produced for surface permanent magnet motors (MOTOR.TYPE = 0).

--> MOTOR.SUPPORTEDPARAMS

```
MOTOR.CTF0
MOTOR.ICONT
MOTOR.INERTIA
MOTOR.IPEAK
MOTOR.KE
MOTOR.KT
MOTOR.LISAT
MOTOR.LQLL
MOTOR.PHASE
MOTOR.POLES
MOTOR.R
MOTOR.VMAX
MOTOR.VOLTMAX
```

Related Topics

7.1 Motor | MOTOR.TYPE (→ p. 629)

19.20.29 MOTOR.TBRAKEAPP

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | The delay time used for applying the motor brake. |
| Units | ms |
| Range | 0 to 1,000 ms |
| Default Value | 75 ms |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 366Eh/0 | M_01-00-00-000 |

Description

This parameter is used to configure the mechanical delay required for the motor brake to close. The AKD PDMM will servo to hold position for the time set, allowing the brake to apply.

NOTE

If MOTOR.AUTOSET = 1 and FB1.MEMVER > 0.01, this parameter will be automatically configured for Kollmorgen motors.

Related Topics

7.1 Motor

19.20.30 MOTOR.TBRAKERLS

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | The delay time used for releasing the motor brake. |
| Units | ms |
| Range | 0 to 1,000 ms |
| Default Value | 75 ms |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 366Fh/0 | M_01-00-00-000 |

Description

This parameter is used to configure the mechanical delay when releasing the motor brake. MOTOR.TBRAKERLS is a time delay that is applied when a brake exists and the drive is enabled. When the drive is enabled, the brake is commanded to release and, during the MOTOR.TBRAKERLS period of time, the drive does not accept a motion command. This delay allows the brake to fully release before the drive begins a new motion.

NOTE

If MOTOR.AUTOSSET = 1 and FB1.MEMVER > 0.01, this parameter will be automatically configured for Kollmorgen motors.

Related Topics

7.1 Motor

19.20.31 MOTOR.TBRAKETO

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Brake apply timeout for vertical axis. |
| Units | Milliseconds |
| Range | -1 to 30,000 |
| Default Value | -1 (function disabled) |
| Data Type | Integer |
| See Also | CS.VTHRESH (→ p. 332), CS.TO (→ p. 331), DRV.DISTO (→ p. 388) |
| Start Version | 01-05-07-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 51EFh/0 | M_01-05-07-000 |

Description

When a drive is disabled (due to user command, digital input, or fault), the brake will normally not be applied until velocity has fallen below CS.VTHRESH (→ p. 332). In some instances, such as a vertical axis, it may be desirable to apply the brake regardless of velocity.

MOTOR.TBRAKETO sets the maximum time allowed to pass between drive disable and application of the motor brake. After this time, the brake will be applied even if velocity is higher than CS.VTHRESH (→ p. 332).

To disable the timer, set the value to -1.

NOTE

Prior to version 01-05-07-000, this timeout was only applied when the Hardware Enable was deactivated and the default was 30,000. From 01-05-07-000 on this timeout is applied in all conditions and the default is -1.

19.20.32 MOTOR.TEMP

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the motor temperature represented as the resistance of the motor PTC or NTC. |
| Units | Ω |
| Range | 0 to $2^{32} \Omega$ |
| Default Value | N/A |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3612h/0 | M_01-00-00-000 |

Description

This parameter returns the motor temperature, which is represented as the resistance of the motor's PTC or NTC.

Related Topics

Motor (→ p. 50) | MOTOR.RTYPE (→ p. 620)

19.20.33 MOTOR.TEMPC

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the motor temperature in degrees Celsius. |
| Units | Degrees Celsius |
| Range | -215 to 215 |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-10-04-000 |

Description

When the motor has a thermistor with linear characteristics (MOTOR.RTYPE = 2, 3 or 4), MOTOR.TEMPC returns the motor temperature in degrees Celsius, which is calculated from the resistance of the motor thermistor.

There is no algorithm to convert from resistance to degrees Celsius for non-linear thermistors. In this case, MOTOR.TEMPC returns -32,768 which is an invalid temperature.

Related Topics

7.1 Motor | MOTOR.TEMP (→ p. 625) | MOTOR.RTYPE (→ p. 620)

19.20.34 MOTOR.TEMPFAULT

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the motor temperature fault level. |
| Units | Ω |
| Range | 0 to 2,000,000,000 Ω |
| Default Value | 0 Ω = switched off |
| Data Type | Integer |
| See Also | MOTOR.TEMP (→ p. 625) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3586h/0 | M_01-00-00-000 |

Description

This parameter is used to configure the motor temperature fault level as a resistance threshold of the motor thermal sensor.

A zero value prevents any fault from being issued.

NOTE

For firmware versions 01-09-00-000 and later, this parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSSET = 1.

Related Topics

7.1 Motor

19.20.35 MOTOR.TEMPWARN

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the motor temperature warning level. |
| Units | Ω |
| Range | 0 to 2,000,000,000 Ω |
| Default Value | 0 Ω = switched off |
| Data Type | Integer |
| See Also | MOTOR.TEMP (→ p. 625) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3453h/0 | M_01-00-00-000 |

Description

This parameter is used to configure the motor temperature warning level as a resistance threshold of the motor PTC.

A zero value prevents any warning from being created.

Related Topics

7.1 Motor

19.20.36 MOTOR.TYPE

| General Information | |
|---------------------|----------------------|
| Type | NV Parameter |
| Description | Sets the motor type. |
| Units | N/A |
| Range | 0, 1, 2, 4, 8 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5104h/0 | M_01-00-00-000 |

Description

MOTOR.TYPE sets the drive control algorithms to different motor types as follows:

0 = Rotary motor

1 = Linear motor

2 = Induction Motor V/f Open Loop Control

4 = Induction Motor Closed Loop Control

8 = Rotary brush motor

The following table shows which MOTOR parameters must be configured for either permanent magnet, induction motor v/f open loop, or induction motor closed loop motor types.

NOTE

ECCN 3A255 imposes special limitations on operating speed for induction motors. See ECCN 3A225 Limitations for Induction Motors (→ p. 1).

| Keyword | Permanent Magnet | Induction Motor v/f Open Loop | Induction Motor Closed Loop |
|---------------------------|------------------|-------------------------------|-----------------------------|
| MOTOR.NAME (→ p. 613) | Yes | Yes | Yes |
| MOTOR.TYPE | Yes | Yes | Yes |
| MOTOR.AUTOSSET (→ p. 592) | Yes | No | Yes |
| MOTOR.IPEAK (→ p. 607) | Yes | Yes | Yes |
| MOTOR.ICONT (→ p. 600) | Yes | Yes | Yes |
| MOTOR.INERTIA (→ p. 606) | Yes | Yes | Yes |
| MOTOR.KT (→ p. 609) | Yes | No | No |
| MOTOR.LQLL (→ p. 612) | Yes | No | Yes |

| Keyword | Permanent Magnet | Induction Motor v/f Open Loop | Induction Motor Closed Loop |
|----------------------------|------------------|-------------------------------|-----------------------------|
| MOTOR.POLES (→ p. 618) | Yes | Yes | Yes |
| MOTOR.VMAX (→ p. 631) | Yes | Yes | Yes |
| MOTOR.R (→ p. 619) | Yes | Yes | Yes |
| MOTOR.VOLTMAX (→ p. 632) | Yes | Yes | Yes |
| MOTOR.PHASE (→ p. 614) | Yes | No | No |
| MOTOR.CTF0 (→ p. 598) | Yes | Yes | Yes |
| MOTOR.KE (→ p. 608) | Yes | No | No |
| MOTOR.IMTR (→ p. 604) | No | No | Yes |
| MOTOR.IMID (→ p. 603) | No | No | Yes |
| MOTOR.VOLTRATED (→ p. 634) | No | Yes | No |
| MOTOR.VRATED (→ p. 635) | No | Yes | No |
| MOTOR.VOLTMIN (→ p. 633) | No | Yes | No |

Related Topics

Motor (→ p. 50)

19.20.37 MOTOR.VMAX

| General Information | |
|---------------------|-------------------------------|
| Type | NV Parameter |
| Description | Sets the maximum motor speed. |
| Units | rpm |
| Range | 100 to 40,000 rpm |
| Default Value | 3,000 rpm |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 35A3h/0 | M_01-00-00-000 |

Description

This parameter sets the maximum motor mechanical speed. The value $1.2 \times \text{MOTOR.VMAX}$ becomes the overall overspeed fault threshold (VL.VFTHRESH) if it is less than the user overspeed fault threshold (VL.THRESH), and for MOTOR.TYPE = 6 if it is less than the motor back EMF overspeed threshold ($(1.02 \times 1000 \times \text{VBUS.OVFTHRESH}) / (\text{MOTOR.KE} \times \sqrt{2})$).

Related Topics

7.1 Motor | VL.THRESH (→ p. 753) | VL.VFTHRESH (→ p. 755)

19.20.38 MOTOR.VOLTMAX

| General Information | |
|---------------------|---------------------------------|
| Type | NV Parameter |
| Description | Sets the motor maximum voltage. |
| Units | Vrms |
| Range | 110 to 900 Vrms |
| Default Value | 230 Vrms |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3452h/0 | M_01-00-00-000 |

Description

This parameter sets the rated nominal operating voltage for the motor insulation system in Vrms. For instance, if a motor rated for a 480 Vac supply is connected to the drive, then MOTOR.VOLTMAX should be set to 480.

This value also sets regen resistor and over voltage thresholds in the drive to acceptable values for the motor so that the motor windings are not damaged. Since VOLTMAX is specified as nominal RMS Voltage, the drive will fault if the DC bus VBUS.VALUE exceeds 2x VOLTMAX which provides sufficient margin to be safe and to prevent false trips.

Related Topics

7.1 Motor

19.20.39 MOTOR.VOLTMIN

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the minimum voltage for V/f control. |
| Units | % |
| Range | 0 to 100% |
| Default Value | 2% |
| Data Type | U16 |
| See Also | MOTOR.VRATED (→ p. 635), MOTOR.VOLTRATED (→ p. 634) |
| Start Version | |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3457h/3 | M_01-00-00-000 |

Description

This parameter configures the drive for the induction motor's minimum voltage at standstill. It is given as a percentage (%) of the motor's rated voltage. MOTOR.VOLTMIN is used to calculate the constant volts per Hertz characteristics of the drive and motor and should be set to a value that generates a current of about 40% of the rated current at standstill.

Related Topics

7.1 Motor

19.20.40 MOTOR.VOLTRATED

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the motor rated voltage. |
| Units | V |
| Range | 50 to 1,000 V |
| Default Value | 230 V |
| Data Type | U16 |
| See Also | MOTOR.VRATED (→ p. 635), MOTOR.VOLTMIN (→ p. 633) |
| Start Version | M_01-03-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3457h/2 | M_01-03-00-000 |

Description

This parameter configures the drive for the induction motor's rated voltage as indicated on the nameplate. MOTOR.VOLTRATED is used to calculate the constant Volts per Hertz characteristics of the drive and motor.

NOTE

If MOTOR.AUTOSET = 1 and FB1.MEMVER > 0.01, this parameter will be automatically configured for Kollmorgen motors.

Related Topics

Motor (→ p. 50)

19.20.41 MOTOR.VRATED

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the motor rated velocity (not maximum velocity) |
| Units | Depends on UNIT.VROTARY (pg 1) or UNIT.VLINEAR (pg 1) Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: counts/s, mm/s, µm/s, custom units/s |
| Range | Rotary: 0.000 to 15,000.000 rpm 0.000 to 250.000 rps 0.000 to 90,000.000 deg/s 0.000 to 1,250.000 custom units/s 0.000 to 1,570.796 rad/s Linear: 0.000 to 1,073,741,824,000.000 counts/s 0.000 to 8,000.000 mm/s 0.000 to 8,000,000.000 µm/s 0.000 to 1,250.000 custom units/s |
| Default Value | 0 rpm |
| Data Type | U16 |
| See Also | MOTOR.VOLTRATED (→ p. 634), MOTOR.VOLTMIN (→ p. 633) |
| Start Version | M_01-03-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3457h/1 | M_01-03-00-000 |

Description

This parameter configures the drive for the rated velocity of the induction motor as indicated on the nameplate. MOTOR.VRATED is used to calculate the constant volts per Hertz characteristics of the drive and motor.

NOTE

If MOTOR.AUTOSSET = 1 and FB1.MEMVER > 0.01, this parameter will be automatically configured for Kollmorgen motors.

Related Topics

Motor (→ p. 50)

19.21 PL Parameters

This section describes the PL parameters.

| | |
|------------------------------------|------------|
| 19.21.1 PL.CMD | 637 |
| 19.21.2 PL.ERR | 638 |
| 19.21.3 PL.ERRFTHRESH | 639 |
| 19.21.4 PL.ERRMODE | 641 |
| 19.21.5 PL.ERRWTHRESH | 643 |
| 19.21.6 PL.FB | 644 |
| 19.21.7 PL.FBSOURCE | 645 |
| 19.21.8 PL.FILTERTIME | 646 |
| 19.21.9 PL.INTINMAX | 647 |
| 19.21.10 PL.INTOUTMAX | 649 |
| 19.21.11 PL.KI | 651 |
| 19.21.12 PL.KITHRESH | 652 |
| 19.21.13 PL.KP | 653 |
| 19.21.14 PL.PDELAY | 654 |

19.21.1 PL.CMD

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the position command directly from the entry to the position loop. |
| Units | Depends on UNIT.PLINEAR (pg 1) or UNIT.PROTARY (pg 1) Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, μ m, (custom units), 16-bit counts |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| See Also | PL.FB (→ p. 644) |
| Start Version | M_01-00-00-000 |

Description

PL.CMD reads the position command as it is received in the position loop entry.

Related Topics

PL.ERR (→ p. 638) | PL.ERRFTHRESH (→ p. 639) | PL.ERRMODE (→ p. 641) | PL.ERRWTHRESH (→ p. 643)

1 Position Loop

19.21.2 PL.ERR

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the position error present when the drive is controlling the position loop. |
| Units | counts, rad, deg, (custom units) |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| See Also | PL.FB (→ p. 644) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|--------------------|----------------------|
| EtherCAT COE and CANopen | 35C5h/0 60F4h/0 | M_01-00-00-000 |

Description

PL.ERR reads the position error present when the drive is controlling the position loop. PL.ERR is the difference between the actual position of the motor shaft (PL.FB (→ p. 644)) and the commanded position of the drive (PL.CMD (→ p. 637)). If the drive is not in the position operating mode (DRV.OPMODE (→ p. 430) = 2), then the PL.ERR value is not generated by the drive and this parameter is read as 0.

Related Topics

PL.ERRFTHRESH (→ p. 639) | PL.ERRMODE (→ p. 641) | PL.ERRWTHRESH (→ p. 643)

1 Position Loop

19.21.3 PL.ERRFTHRESH

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the maximum position error. |
| Units | Depends UNIT.ACCROTARY (pg 1) on or UNIT.ACCLINEAR (pg 1) Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, μ m, (custom units), 16-bit counts |
| Range | Rotary: 0.000 to 5,123,372,000,000,005.000 counts 0.000 to 7,495,067.136 rad 0.000 to 429,436,076.032 deg 0.000 to 5,964,389.888 (custom units) 0.000 to 78,176,452,636.718 16-bit counts Linear: 0.000 to 5,123,372,000,000,005.000 counts 0.000 to 1,192,877.952*MOTOR.PITCH (\rightarrow p. 617) mm 0.000 to 1,192,878,014.464*MOTOR.PITCH (\rightarrow p. 617) μ m 0.000 to 5,964,389.888 (custom units) 0.000 to 78,176,452,636.718 16-bit counts |
| Default Value | Rotary: 42,949,672,960.000 counts 62.832 rad 3,600.000 deg 50.000 (custom units) 655,360.000 16-bit counts Linear: 42,949,672,960.000 counts 10.000*MOTOR.PITCHMOTOR.PITCH (\rightarrow p. 617)mm 10,000.000*MOTOR.PITCH μ m 50.000 (custom units) 655,360.000 16-bit counts |
| Data Type | Float |
| See Also | PL.ERR |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 35C7h/0 | M_01-00-00-000 |
| | 6065h/0 | |

Description

This parameter sets the maximum position error. If the position error PL.ERR (\rightarrow p. 638) is larger than PL.ERRFTHRESH the drive generates a fault. If PL.ERRFTHRESH is set to 0, the maximum position error is ignored.

Example

Set position rotary units to 2 (degrees). Setting PL.ERRFTHRESH to 1000 states that if the position error is larger than 1000 degrees, the drive will generate a fault.

UNIT.PROTARY 2

PL.ERRFTHRESH 1000

Related Topics

PL.ERR (→ p. 638) | PL.ERRMODE (→ p. 641) | PL.ERRWTHRESH (→ p. 643)

1 Position Loop

19.21.4 PL.ERRMODE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the type of following error warning and fault usage. |
| Units | 0- Standard following error 1-Enhanced following error |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Boolean |
| See Also | PL.ERR (→ p. 638), PL.ERRFTHRESH (→ p. 639), PL.ERRWTHRESH (→ p. 643) |
| Start Version | M_01-02-09-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5121h/0 | M_01-02-09-000 |

Description

PL.ERRMODE sets the type of following error warning and fault usage.

Mode 0 - following error magnitude fault

In Mode 0, the values of PL.ERRFTHRESH and PL.ERRWTHRESH are compared against the value of PL.ERR. If the absolute value of PL.ERR is larger than PL.ERRWTHRESH, then a warning is generated. If the absolute value of PL.ERR is larger than PL.ERRFTHRESH, then a fault is generated.

Mode 1 - deviation from predicted trajectory fault

In Mode 1, the values of PL.ERRFTHRESH and PL.ERRWTHRESH are compared against the following value:

$$\langle \text{error} \rangle = \text{abs}(\text{PL.ERR} - [(\text{VL.CMD} - \text{VL.FF}) / \text{PL.KP}])$$

If the absolute value of $\langle \text{error} \rangle$ is larger than PL.ERRWTHRESH for a consecutive period of 100 ms, then a warning is generated. If the absolute value of $\langle \text{error} \rangle$ is larger than PL.ERRFTHRESH for a consecutive period of 100 ms, then a fault is generated.

In mode 1, if PL.KI is not 0 then the following error prediction mechanism is turned off. When the drive is disabled, the following error limit tests are turned off and the warnings are cleared. A value of 0 in PL.ERRFTHRESH or PL.ERRWTHRESH disables the respective functionality.

Usage of Mode 1:

If a machine was frictionless, no resonances and VL.KVFF was set to 1.0 (100% velocity feedforward) then the value being compared against the fault threshold and warning threshold would be zero because all of the error would be due to the commanded move.

This feature is intended to remove all error contributed from the trajectory and leave only the disturbances from forces outside of those commanded.

Example

Assuming

PL.ERRMODE = 0, PL.ERRFTHRESH=1.2, PL.ERRWTHRESH=1, then PL.ERR reads 1.1.

In this case the warning is generated, but the fault is not.

Assuming PL.ERRMODE = 0, PL.ERRFTHRESH=1.2, PL.ERRWTHRESH=1, then PL.ERR reads 1.3.

In this case the warning is generated, as well as the fault.

Related Topics

[PL.ERR \(→ p. 638\)](#) | [PL.ERRFTHRESH \(→ p. 639\)](#) | [PL.ERRWTHRESH \(→ p. 643\)](#)

1 Position Loop

19.21.5 PL.ERRWTHRESH

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the position error warning level. |
| Units | Depends on UNIT.PLINEAR (pg 1) or UNIT.PROTARY (pg 1) Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, μm , (custom units), 16-bit counts |
| Range | Rotary: 0.000 to 5,123,372,000,000,005.000 counts 0.000 to 7,495,067.136 rad 0.000 to 429,436,076.032 deg 0.000 to 5,964,389.888 (custom units) 0.000 to 78,176,452,636.718 16-bit counts Linear: 0.000 to 5,123,372,000,000,005.000 counts 0.000 to 1,192,877.952*MOTOR.PITCH (\rightarrow p. 617) mm 0.000 to 1,192,878,014.464*MOTOR.PITCH (\rightarrow p. 617) μm 0.000 to 5,964,389.888 (custom units) 0.000 to 78,176,452,636.718 16-bit counts |
| Default Value | 0.000 deg |
| Data Type | Float |
| See Also | PL.ERR (\rightarrow p. 638) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3483h/0 | M_01-00-00-000 |

Description

If this value is not equal 0 and the position error PL.ERR (\rightarrow p. 638) is larger than this value, the drive will generate a warning.

If PL.ERRWTHRESH is set to 0 the warning is not issued.

Example

Set position rotary units to 2 degrees. If you set PL.ERRWTHRESH to 100 and the position error is larger than 100 degrees, then the drive will generate a warning.

```
UNIT.PROTARY 2
```

```
PL.ERRWTHRESH 100
```

Related Topics

PL.ERR (\rightarrow p. 638) | PL.ERRFTHRESH (\rightarrow p. 639) | PL.ERRMODE (\rightarrow p. 641)

1 Position Loop

19.21.6 PL.FB

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the position feedback value. |
| Units | Depends on UNIT.PLINEAR (pg 1) or UNIT.PROTARY (pg 1) Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, μ m, (custom units), 16-bit counts |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| See Also | FB1.OFFSET |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 6064h/0 | M_01-00-00-000 |

Description

PL.FB returns the position feedback value.

Note that this value is not the pure feedback value read from the feedback device, but also includes the value of the FB1.OFFSET and an internal offset set automatically by the FW when a homing switch is actuated.

Related Topics

PL.ERR (→ p. 638) | PL.ERRFTHRESH (→ p. 639) | PL.ERRMODE (→ p. 641) | PL.ERRWTHRESH (→ p. 643)

1 Position Loop

19.21.7 PL.FBSOURCE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the feedback source for the position loop. |
| Units | N/A |
| Range | Range will differ depending on drive model. 0 to 1 (for AKD PDMM-x-xxxxx-NAxx-xxxx) 0 to 2 (for AKD PDMM-x-xxxxx-NBxx-xxxx) |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5128h/0 | M_01-00-00-000 |

Description

This parameter determines the feedback source that the position loop uses. A value of 0 for this parameter selects the primary feedback, a value of 1 selects the secondary feedback. If you use the secondary feedback as the source for the position loop, then FB2.MODE mode should be set as 0 (A/B signals). A/B signals are the only supported feedback type as secondary feedback into the position loop. Other settings for FB2.MODE are intended as pulse inputs or a gearing command when PL.FBSOURCE remains 0.

| | |
|---|---|
| 0 | Primary Feedback connected to X10. |
| 1 | Secondary Feedback (DRV.HANDWHEEL) connected to X7 or X9. |
| 2 | Tertiary Feedback connected to X9 (only supported with AKD PDMM-x-xxxxx-NBxx-xxxx). |

Related Topics

PL.ERR (→ p. 638) | PL.ERRFTHRESH (→ p. 639) | PL.ERRMODE (→ p. 641) | PL.ERRWTHRESH (→ p. 643)

1 Position Loop

19.21.8 PL.FILTERTIME

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the position command filter period. |
| Units | Milliseconds |
| Range | 0, 0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512 |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-08-01-000 |

Description

PL.FILTERTIME sets the period of a "boxcar" moving average filter on the position command. When PL.FILTERTIME is greater than zero, the filter is active.

The moving average filter is applied to all Command Source types, but only active when the AKD PDMM is in Position mode.

This feature is typically used for smoothing abrupt changes in position command, or filtering coarse command steps from low resolution controllers, or electronic gearing.

NOTE

When the filter is on (PL.FILTERTIME > 0), total motion commanded will be delayed by the period the filter is set to.

19.21.9 PL.INTINMAX

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Limits the input of the position loop integrator by setting the input saturation. |
| Units | Depends on UNIT.PLINEAR (pg 1) or UNIT.PROTARY (pg 1) Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, μm , (custom units), 16-bit counts |
| Range | Rotary: 0.000 to 18,446,744,073,709.000 counts 0.000 to 26,986.052 rad 0.000 to 1,546,188.288 deg 0.000 to 21,474.836 (custom units) 0.000 to 281,474,976.710 16-bit counts Linear: 0.000 to 18,446,744,073,709.000 counts 0.000 to 4,294.968*MOTOR.PITCH (\rightarrow p. 617) mm 0.000 to 4,294,967.296*MOTOR.PITCH (\rightarrow p. 617) μm 0.000 to 21,474.836 (custom units) 0.000 to 281,474,976.710 16-bit counts |
| Default Value | Rotary: 3,999,989,760.000 counts 5.852 rad 335.275 deg 4.657 (custom units) 61,035.000 16-bit counts Linear: 3,999,989,760.000 counts 0MOTOR.PITCH (\rightarrow p. 617) mm 9MOTOR.PITCH (\rightarrow p. 617) μm 4.657 (custom units) 61,035.000 16-bit counts |
| Data Type | Float |
| See Also | PL.FB |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3481h/1 | M_01-00-00-000 |

Description

PL.INTINMAX limits the input of the position loop integrator by setting the input saturation. When used in concert with PL.INSATOUT, this variable allows you to make the position loop integrator effective near the target position. Far from the target position, however, the integrator is not dominant in the loop dynamics.

Related Topics

PL.ERR (→ p. 638) | PL.ERRFTHRESH (→ p. 639) | PL.ERRMODE (→ p. 641) | PL.ERRWTHRESH (→ p. 643) | Position Loop (→ p. 1)

19.21.10 PL.INTOUTMAX

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Limits the output of the position loop integrator by setting the output saturation. |
| Units | Depends on UNIT.PLINEAR (pg 1) or UNIT.PROTARY (pg 1) Rotary: counts, rad, deg, (custom units), 16-bit counts Linear: counts, mm, μm , (custom units), 16-bit counts |
| Range | Rotary: 0.000 to 18,446,744,073,709.000 counts 0.000 to 26,986.052 rad 0.000 to 1,546,188.288 deg 0.000 to 21,474.836 (custom units) 0.000 to 281,474,976.710 counts 16 bit Linear: 0.000 to 18,446,744,073,709.000 counts 0.000 to 4,294.968*MOTOR.PITCH (\rightarrow p. 617) mm 0.000 to 4,294,967.296*MOTOR.PITCH μm 0.000 to 21,474.836 (custom units) 0.000 to 281,474,976.710 16-bit counts |
| Default Value | Rotary: 3,999,989,760.000 counts 5.852 rad 335.275 deg 4.657 (custom units) 61,035.000 16-bit counts Linear: 3,999,989,760.000 counts 0MOTOR.PITCH (\rightarrow p. 617) mm 9MOTOR.PITCH (\rightarrow p. 617) μm 4.657 (custom units) 61,035.000 16-bit counts |
| Data Type | Float |
| See Also | PL.INTINMAX |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3481h/2 | M_01-00-00-000 |

Description

PL.INTOUTMAX limits the output of the position loop integrator by setting the output saturation.

When used in concert with PL.INTINMAX, this variable allows you to make the position loop integrator effective near the target position. Far from the target position, however, the integrator is not dominant in the loop dynamics.

Related Topics

PL.ERR (→ p. 638) | PL.ERRFTHRESH (→ p. 639) | PL.ERRMODE (→ p. 641) | PL.ERRWTHRESH (→ p. 643)

1 Position Loop

19.21.11 PL.KI

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the integral gain of the position loop. |
| Units | Hz |
| Range | 0 to 250 Hz |
| Default Value | 0 Hz |
| Data Type | Float |
| See Also | PL.KP, PL.KD |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3480h/0 | M_01-00-00-000 |

Description

PL.KI sets the integral gain of the position regulator PID loop.

Related Topics

PL.ERR (→ p. 638) | PL.ERRFTHRESH (→ p. 639) | PL.ERRMODE (→ p. 641) | PL.ERRWTHRESH (→ p. 643)

1 Position Loop

19.21.12 PL.KITHRESH

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the velocity command threshold at which the position loop integrator turns off. |
| Units | Velocity (See UNIT.VROTARY, UNIT.VLINEAR) |
| Range | 0 to VL.LIMITP |
| Default Value | VL.LIMITP |
| Data Type | Float |
| Start Version | M_01-14-01-000 |

Description

This parameter sets the velocity command threshold when the position loop integrator turns off.

If velocity command is less than PL.KITHRESH, the position loop integrator will be active. If the velocity command exceeds PL.KITHRESH the position loop integrator is disabled (it will act as if PL.KI (→ p. 651) = 0).

Common Uses

As a general rule, PL.KI should not be used at the same time as VL.KI (→ p. 740), the two integrators will oppose each other and likely cause undesirable results, sometimes a runaway. Using PL.KITHRESH in conjunction with VL.KIMODE the user can specify operating conditions to switch between integrators based on machine operation.

It often makes sense to use the velocity integrator during a move, but as the motion comes to an end, to switch back to the position integrator for fine positioning.

Related Topics

PL.KI (→ p. 651) | VL.KIMODE (→ p. 741)

1 Position Loop

19.21.13 PL.KP

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the proportional gain of the position regulator PID loop. |
| Units | (rev/s)/rev |
| Range | 0 to 2,147,483.008 (rev/s)/rev |
| Default Value | 100 rps/rev |
| Data Type | Float |
| See Also | PL.KI (→ p. 651), |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3542h/0 | M_01-00-00-000 |

Description

PL.KP sets the proportional gain of the position regulator PID loop.

Related Topics

PL.ERR (→ p. 638) | PL.ERRFTHRESH (→ p. 639) | PL.ERRMODE (→ p. 641) | PL.ERRWTHRESH (→ p. 643)

1 Position Loop

19.21.14 PL.PDELAY

| General Information | |
|---------------------|---------------------------------|
| Type | R/W Parameter |
| Description | Sets the delay time for PL.CMD. |
| Units | Microseconds |
| Range | 0, 100 to 625,000 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-13-01-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 53D5/0 | M_01-13-01-000 |

Description

PL.PDELAY inserts a delay before the position command signal. The delay is implemented after the feed-forwards are calculated, which allows the feedforwards to “lead” the position command/error signals.

By tuning this filter, position loop overshoot can be reduced without affecting loop stability. This filter is considered a feedforward. For best results it should be set in conjunction with VL.FFDELAY.

After making an approximation of the Open Loop bandwidth (Hz) in the Performance Servo Tuner, calculate PL.PDELAY:

$$\text{PL.PDELAY} = 1 / (\text{Open Loop BW (Hz)} * 2\pi) * 1,000,000$$

NOTE

1,000,000 = seconds to microseconds

NOTE

PL.PDELAY and VL.FFDELAY are calculated exactly the same, however, depending on the mode of operation, sometimes an additional 250us should be added to PL.PDELAY for optimal results.

NOTE

If PL.PDELAY is not zero, and the drive is in Field Bus mode, then commanding position jumps above one revolution within one fieldbus cycle are not supported and can result in undefined motion.

Related Topics

Using the Performance Servo Tuner (→ p. 104)

VL.FFDELAY (→ p. 737)

1 Position Loop

19.22 REC Parameters

This section describes the REC parameters.

| | |
|---|------------|
| 19.22.1 REC.ACTIVE | 656 |
| 19.22.2 REC.CH1 to REC.CH6 | 657 |
| 19.22.3 REC.DONE | 658 |
| 19.22.4 REC.GAP | 659 |
| 19.22.5 REC.NUMPOINTS | 660 |
| 19.22.6 REC.OFF | 661 |
| 19.22.7 REC.RECPRMLIST | 662 |
| 19.22.8 REC.RETRIEVE | 663 |
| 19.22.9 REC.RETRIEVEDATA | 664 |
| 19.22.10 REC.RETRIEVEFRMT | 666 |
| 19.22.11 REC.RETRIEVEHDR | 667 |
| 19.22.12 REC.RETRIEVESIZE | 668 |
| 19.22.13 REC.STOPTYPE | 669 |
| 19.22.14 REC.TRIG | 670 |
| 19.22.15 REC.TRIGPARAM | 671 |
| 19.22.16 REC.TRIGPOS | 672 |
| 19.22.17 REC.TRIGPRMLIST | 674 |
| 19.22.18 REC.TRIGSLOPE | 675 |
| 19.22.19 REC.TRIGTYPE | 676 |
| 19.22.20 REC.TRIGVAL | 677 |

19.22.1 REC.ACTIVE

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Indicates if data recording is in progress (active). |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | N/A |
| Data Type | Integer |
| See Also | REC.DONE (→ p. 658), REC.OFF (→ p. 661) |
| Start Version | M_01-00-00-000 |

Description

REC.ACTIVE indicates whether or not data recording is in progress. Recording is in progress if the trigger was met and the recorder is recording all data.

Related Topics

Scope (→ p. 154)

19.22.2 REC.CH1 to REC.CH6

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets recording channels 1 to 6. |
| Units | N/A |
| Range | N/A |
| Default Value | CH1 = IL.FB CH2 = IL.CMD CH3 = VL.FB CH4 = Empty CH5 = Empty CH6 = Empty |
| Data Type | String |
| See Also | REC.TRIG (→ p. 670) |
| Start Version | M_01-00-00-000 |

Description

REC.CHx specifies the recording channels.

There are 3 options to set the recording channels values:

- Set 0, CLR, or CLEAR. This setting clears the recording channel.
- Set one of the recordable commands. The list of recordable commands can be obtain by executing REC.RECPRMLIST (→ p. 662).
- Set an internal value or variable of the drive (same as for DRV.MEMADDR (→ p. 421) input).

Related Topics

Scope (→ p. 154)

19.22.3 REC.DONE

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Checks whether or not the recorder has finished recording. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | N/A |
| Data Type | Integer |
| See Also | REC.ACTIVE, REC.OFF |
| Start Version | M_01-00-00-000 |

Description

REC.DONE indicates that the recorder has finished recording. This value is reset to 0 when the recorder trigger is set. The drive also resets this value when the recording has finished or when REC.OFF is executed.

Related Topics

Scope (→ p. 154)

19.22.4 REC.GAP

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Specifies the gap between consecutive samples. |
| Units | N/A |
| Range | 1 to 65,535 |
| Default Value | 1 |
| Data Type | Integer |
| See Also | REC.TRIG (→ p. 670) |
| Start Version | M_01-00-00-000 |

Description

REC.GAP specifies the gap between consecutive samples. The recording base rate is 16 kHz, thus a gap of 1 means that a sample is recorded every 62.5 μ s.

Related Topics

Scope (→ p. 154)

19.22.5 REC.NUMPOINTS

| General Information | |
|---------------------|--------------------------------------|
| Type | R/W Parameter |
| Description | Sets the number of points to record. |
| Units | N/A |
| Range | 1 to 65,535 |
| Default Value | 1,000 |
| Data Type | Integer |
| See Also | REC.TRIG (→ p. 670) |
| Start Version | M_01-00-00-000 |

Description

REC.NUMPOINTS specifies the number of points (samples) to record.

Related Topics

Scope (→ p. 154)

19.22.6 REC.OFF

| General Information | |
|---------------------|-------------------------|
| Type | R/W Parameter |
| Description | Turns the recorder OFF. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| See Also | REC.ACTIVE, REC.DONE |
| Start Version | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

REC.OFF turns the recorder off. In order to set the recorder again, the recorder must first be armed and then a trigger set.

Related Topics

Scope (→ p. 154)

REC.READY

19.22.7 REC.RECPRMLIST

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the list of recordable parameters. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| See Also | REC.CH1 to REC.CH6 (→ p. 657) |
| Start Version | M_01-00-00-000 |

Description

This command returns the list of recordable parameters. You can use a recordable parameter as an input to any of the recording channels.

Note that an internal address or a registered variable can be used as input to any of the channels in addition to the list.

Related Topics

Scope (→ p. 154)

19.22.8 REC.RETRIEVE

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Transfers all the recorded data to the communication channel. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | String |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

REC.RETRIEVE causes the drive to transfer all the recorded data to the communication channel.

Example

The following format is the retrieve reply format (for N samples, G sample gap, and M parameters, where $M \leq 6$):

```
Recording
<N>, <G>
<parameter name 1> ... <parameter name M>
Value11 ... Value1M
Value N1 ... ValueNM
```

Related Topics

Scope (→ p. 154)

19.22.9 REC.RETRIEVEDATA

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Retrieves the recorded data without the header. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| See Also | REC.RETRIEVE, REC.RETRIEVEHDR, REC.RETRIEVESIZE |
| Start Version | M_01-00-00-000 |

Description

REC.RETRIEVEDATA retrieves a section of recorded data according to REC.RETRIEVESIZE from the received index; if no index is received, the drive retrieves the data from next section. An index is supplied to enable multiple retrieves and to give better control on the buffer in case of overflow. If no index or a negative value is present, then the index is ignored.

EWV¹ uses this parameter to retrieve the data continuously for real time recoding.

The size of the data returned by this command depends on the number set by REC.RETRIEVESIZE.

Use REC.RETRIEVE for complete recording information view.

Notes:

- If REC.RETRIEVESIZE is larger than the buffer size, then it simply returns the whole buffer (no error).
- If the index is received, the data will be continuously returned starting from the given index (default starting index is 0).
- If the index is out of the bounds of the buffer, then it will be ignored.
- If recorder is active and REC.STOPTYPE==0, then this parameter returns an error.
- If REC.STOPTYPE==1, then this parameter returns the next section of data in the buffer (even if it reached the end of the buffer, it will return to the beginning of the buffer and add the data from index 0.)
- If REC.STOPTYPE==1 and the retrieve is too slow (gets overrun by the recorder), an overflow error message is returned instead of the retrieved data.
- If REC.STOPTYPE==0 and no index is received, continuously send the sections of data until the end of the buffer is reached. Then, return to the beginning of buffer and continue.
- A new REC.TRIG (→ p. 670) command automatically sets the index to 0.

Example

The following example retrieves data from index 100 in the size of 10 (hence places 100 to 109 in the buffer)

```
REC.NUMPOINTS 1000
REC.RETRIVESIZE 10
REC.TRIG
REC.RETRIEVEDATA 100
```

Related Topics

¹Embedded Workbench Views

Scope (→ p. 154)

19.22.10 REC.RETRIEVEFRMT

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the format for recorded data output. |
| Units | N/A |
| Range | 0 to 1; 0 = Standard format, 1 = Internal format (high speed) |
| Default Value | 1 |
| Data Type | Integer |
| See Also | REC.RETRIEVE (→ p. 663), REC.RETRIEVEDATA (→ p. 664) |
| Start Version | M_01-00-00-000 |

Description

Recorded data is transferred to the communication channel in one of two formats: standard or internal high speed. The standard (slower) format cannot be used for continuous recording, but is more easily read. The high speed format allows continuous data recording (needed for autotuning). EWV¹ supports both formats.

Example

The following recorder data is in standard format:

```
10,1
IL.FB,VL.CMD,VL.FB
-0.086,0.000,2.661
0.000,0.000,3.605
0.029,0.000,-0.486
```

The following recorder data is in internal format:

```
10,1
IL.FB,VL.CMD,VL.FB
F3-0x56,F30x0,F30xA65
F30x0,F30x0,F30xE15
F30x1D,F30x0,F3-0x1E6
```

Related Topics

Scope (→ p. 154)

¹Embedded Workbench Views

19.22.11 REC.RETRIEVEHDR

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Retrieves the recorded header without the data. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| See Also | REC.RETRIEVE, REC.RETRIEVEDATA |
| Start Version | M_01-00-00-000 |

Description

This command retrieves the recorded header without the data of the recording.

EWV¹ uses this parameter to retrieve the header once before continuously reading the data for RT recoding.

Use REC.RETRIEVE for complete recording information view.

Related Topics

Scope (→ p. 154)

¹Embedded Workbench Views

19.22.12 REC.RETRIEVESIZE

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the number of samples that REC.RETRIEVEDATA returns. |
| Units | recorder samples |
| Range | 0 to 65,535 recorder samples |
| Default Value | 1,000 recorder samples |
| Data Type | Integer |
| See Also | REC.RETRIEVEDATA (→ p. 664), REC.RETRIEVEHDR (→ p. 667) |
| Start Version | M_01-00-00-000 |

Description

This parameter sets the number of samples that REC.RETRIEVEDATA (→ p. 664) returns.

EWV¹ also uses this parameter to set the number of samples returned when retrieving the data continuously for RT recoding.

Use REC.RETRIEVE (→ p. 663) for the complete recording information view.

Related Topics

Scope (→ p. 154)

¹Embedded Workbench Views

19.22.13 REC.STOPTYPE

| General Information | |
|---------------------|---------------------------------------|
| Type | R/W Parameter |
| Description | Sets the recorder stop type. |
| Units | N/A |
| Range | 0 or 1 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | REC.RETRIEVEDATA, REC.RETRIEVESIZE |
| Start Version | M_01-00-00-000 |

Description

This parameter sets the stop type for the recording.

0 = Recorder fills in the buffer once.

1 = Recorder runs, continuously filling the recording circular buffer.

To stop RT recording, execute REC.OFF.

Related Topics

Scope (→ p. 154)

19.22.14 REC.TRIG

| General Information | |
|---------------------|--------------------------|
| Type | Command |
| Description | Triggers the recorder. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| See Also | REC.RETRIEVE, REC.OFF |
| Start Version | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

REC.TRIG starts the trigger according to the trigger type defined by REC.TRIGTYPE.

REC.TRIG sets the value of REC.DONE to 0.

After calling REC.TRIG, the data that was recorded by previous recording is deleted and cannot be retrieved.

No REC parameters can be set after a call to REC.TRIG until the recorder has finished or until REC.OFF is executed.

Related Topics

Scope (→ p. 154)

19.22.15 REC.TRIGPARAM

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the parameter that triggers the recorder. |
| Units | N/A |
| Range | N/A |
| Default Value | IL.FB |
| Data Type | String |
| See Also | REC.TRIG |
| Start Version | M_01-00-00-000 |

Description

REC.TRIGPARAM sets the parameter on which the recorder triggers.

This parameter is only used when REC.TRIGTYPE = 2.

Input values are:

1. One of the set drive parameters list that can be set as a trigger. The available parameters for trigger are: PL.ERR (→ p. 638), PL.CMD, PL.FB, VL.CMD, VL.FB, IL.CMD, and IL.FB.
2. Internal value or variable of the drive (same as for DRV.MEMADDR input).

Related Topics

Scope (→ p. 154)

19.22.16 REC.TRIGPOS

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the trigger position in the recording buffer. |
| Units | % |
| Range | 1 to 100% |
| Default Value | 10% |
| Data Type | Integer |
| See Also | REC.TRIG (→ p. 670), REC.NUMPOINTS (→ p. 660) |
| Start Version | M_01-00-00-000 |

Description

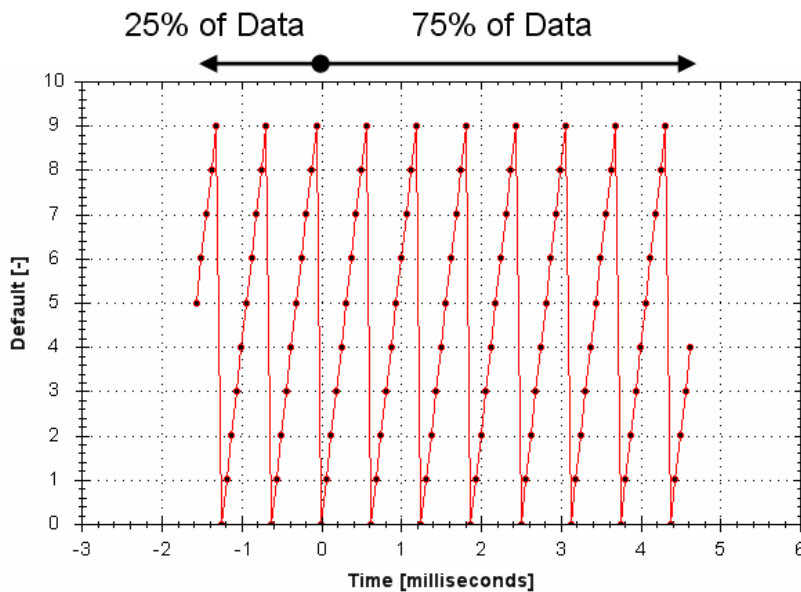
REC.TRIGPOS sets the trigger position in the recording buffer. The recording buffer size is defined by REC.NUMPOINTS. The input value is a percentage of the buffer (that is, a value of 25 means saving 25% of the buffer data before the trigger occurs and 75% after it occurs). This parameter is only used when REC.TRIGTYPE = 2 or 3.

Trigger Position

Trigger position (REC.TRIGPOS) allows you to collect data that occurs before the trigger occurs. In some instances, you may want to see the conditions prior to the trigger. Trigger position lets you control how much signal is collected before the trigger condition occurred.

Trigger position is specified in units of percent (%). If you specify a trigger position of $x\%$, then $x\%$ of the data is before 0 ms in the data time and $100-x\%$ (the rest of the data) is at or greater than 0 ms. In the figure below, the trigger position is set to 25% (REC.TRIGPOS 25).

In the EWV¹ scope, the 0 time point is clear. When collecting the data using REC.RETRIEVE or similar commands, the time is not returned, so some caution should be used when the trigger point is important to understand.



¹Embedded Workbench Views

Related Topics

Scope (→ p. 154)

19.22.17 REC.TRIGPRMLIST

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the list of possible trigger parameters. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| See Also | REC.TRIGPARAM |
| Start Version | M_01-00-00-000 |

Description

This command returns the list of trigger parameters. Each one of those parameters can serve as the trigger parameter (input to REC.TRIGPARAM).

Note that an internal address or a registered variable can be used as input to REC.TRIGPARAM in addition to the list that this parameter returns.

Related Topics

Scope (→ p. 154)

19.22.18 REC.TRIGSLOPE

| General Information | |
|---------------------|------------------------------|
| Type | R/W Parameter |
| Description | Sets the trigger slope. |
| Units | 0 = Negative 1 = Positive |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | REC.TRIG, REC.NUMPOINTS |
| Start Version | M_01-00-00-000 |

Description

REC.TRIGSLOPE sets the recorder trigger slope. This parameter is only used when REC.TRIGTYPE = 2 or 3.

Related Topics

Scope (→ p. 154)

19.22.19 REC.TRIGTYPE

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the trigger type. |
| Units | 0 = immediate 1 = command 2 = parameter 3 = boolean |
| Range | 0 to 2 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | REC.TRIG (→ p. 670), REC.TRIGPARAM, REC.TRIGVAL, REC.TRIGSLOPE REC.TRIGPOS |
| Start Version | M_01-00-00-000 |

Description

REC.TRIGTYPE sets the type of trigger.

Input values are as follows:

| Value | Description |
|-------|---|
| 0 | Recording starts immediately |
| 1 | Recording starts on the next command executed through the TCP/IP. The trigger location in the buffer is set according to REC.TRIGPOS. |
| 2 | Recording starts per the values of REC.TRIGPARAM, REC.TRIGVAL, REC.TRIGSLOPE, and REC.TRIGPOS. |
| 3 | Recording starts when the value of REC.TRIGPARAM is 0 for REC.TRIGSLOPE = 0 or 1 for REC.TRIGSLOPE = 1 |

Related Topics

Scope (→ p. 154)

19.22.20 REC.TRIGVAL

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the trigger value. |
| Units | The units of the parameter are chosen according to the unit type. |
| Range | 0 to 2 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | REC.TRIG (→ p. 670), REC.TRIGPARAM, REC.TRIGVAL, REC.TRIGSLOPE, REC.TRIGPOS |
| Start Version | M_01-00-00-000 |

Description

REC.TRIGVAL is the value that must be met by REC.TRIGPARAM for the trigger to occur. The units of this parameter are set according to the units of REC.TRIGPARAM.

Related Topics

Scope (→ p. 154)

19.23 REGEN Parameters

This section describes the REGEN parameters.

| | |
|--|------------|
| 19.23.1 REGEN.POWER | 679 |
| 19.23.2 REGEN.POWERFILTERED | 680 |
| 19.23.3 REGEN.REXT | 681 |
| 19.23.4 REGEN.TEXT | 682 |
| 19.23.5 REGEN.TYPE | 683 |
| 19.23.6 REGEN.WATTEXT | 684 |

19.23.1 REGEN.POWER

| General Information | |
|---------------------|--|
| Type | R/O parameter |
| Description | Reads regen resistor's calculated power. |
| Units | Watt |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3416h/0 | M_01-00-00-000 |

Description

This parameter reads regen resistor's calculated power, which is determined as follows:

$$(V^2 / R) * DutyCycle$$

Related Topics

6.4 Regeneration

19.23.2 REGEN.POWERFILTERED

| General Information | |
|---------------------|---|
| Type | R/O parameter |
| Description | Returns a filtered version of REGEN.POWER . |
| Units | Watt |
| Range | 0 to REGEN.WATTEXT |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-07-01-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3417h/0 | M_01-00-00-000 |

Description

This parameter returns a filtered value of REGEN.POWER. A single order lowpass of $1/\text{REGEN.TEXT}$ Hz is applied to generate REGEN.POWERFILTERED.

If REGEN.POWERFILTERED exceeds the value of REGEN.WATTEXT, Fault 521 (Regen overpower) will be generated.

This keyword may be recorded in the scope screen for analysis.

Example

REGEN.TEXT = 10 (seconds)

Lowpass Filter Frequency = $1/10$ seconds = 0.1 Hz

REGEN.POWERFILTERED = REGEN.POWER * 0.1Hz lowpass

Related Topics

REGEN.POWER (→ p. 679) | REGEN.TEXT (→ p. 682) | REGEN.WATTEXT (→ p. 684)

19.23.3 REGEN.REXT

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the external, user-defined regen resistor resistance. |
| Units | Ω |
| Range | 0 to 255 Ω |
| Default Value | 0 Ω |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 35C2h/0 | M_01-00-00-000 |

Description

REGEN.REXT sets the external user-defined regen resistor resistance. This variable is needed for the regen resistor temperature estimation algorithm.

Related Topics

6.4 Regeneration

19.23.4 REGEN.TEXT

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the external regen resistor thermal protection time constant. |
| Units | Seconds |
| Range | 0.1 to 1,200 Seconds |
| Default Value | 100 Seconds |
| Data Type | Float |
| See Also | REGEN.WATTEXT (→ p. 684), REGEN.REXT (→ p. 681) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3415h/0 | M_01-00-00-000 |

Description

REGEN.TEXT is the time-to-fault when input power steps from 0 to 150% of REGEN.WATTEXT. REGEN.TEXT sets the thermal time constant of the algorithm used to protect the regeneration resistor from overheating and failing. The drive's regen resistor protection algorithm continuously calculates the power dissipated in the resistor and processes that power value through a single pole low pass filter to model the regen resistor's thermal inertia. When the filtered regen power on the output of the filter (REGEN.POWERFILTERED) exceeds REGEN.WATTEXT, a fault occurs. REGEN.TEXT sets the time constant of this thermal inertial filter = (REGEN.TEXT/1.1).

REGEN.TEXT can often be found directly on power resistor data sheets. On the data sheet find the peak overload curve and then find the safe allowed time to be at 150% of the regen resistor's continuous power rating. Set REGEN.TEXT to this value in seconds. Another way regen resistor peak overload capability is often specified is by giving the energy rating in joules of the resistor. If you have the energy rating joule limit E then:

$$\text{REGEN.TEXT} = (1.1) * ((E \text{ in joules}) / \text{REGEN.WATTEXT})$$

Example

The external regen resistor is rated for 250 W continuous, is 33 ohm, and has a joule rating of 500 joules. To use this resistor, the drive settings become:

REGEN.TYPE = -1 (External Regen)

REGEN.REXT = 33

REGEN.WATTEXT = 250

REGEN.TEXT = (1.1)*(500 j)/(250 W) = 2.2 sec

Related Topics

6.4 Regeneration

19.23.5 REGEN.TYPE

| General Information | |
|---------------------|-------------------------------|
| Type | NV Parameter |
| Description | Sets the regen resistor type. |
| Units | N/A |
| Range | -1 to 0 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3412h/0 | M_01-00-00-000 |

Description

You can specify a user-defined external regen resistor, select an internal regen resistor, or choose from a list of predefined regen resistors. The values for REGEN.TYPE are shown below:

| Type | Description |
|------|--------------------------------------|
| -1 | External user-defined regen resistor |
| 0 | Internal regen resistor |

If you specify a user-defined regen resistor, then you must also define this resistor's resistance (REGEN.REXT), heatup time (REGEN.REXT), and power (REGEN.WATTEXT).

Related Topics

6.4 Regeneration

19.23.6 REGEN.WATTEXT

| General Information | |
|---------------------|---|
| Type | R/W parameter |
| Description | Sets the regen resistor's power fault level for an external regen resistor. |
| Units | W |
| Range | 0 to 62,000 W |
| Default Value | 1000 W |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3414h/0 | M_01-00-00-000 |

Description

Sets the regen resistor's power fault level for an external regen resistor (when REGEN.TYPE = -1). Above this fault level, the regen resistor's PWM will be 0 and a fault will be issued.

Related Topics

6.4 Regeneration

19.24 SM Parameters

This section describes the SM parameters.

| | |
|------------------------------|------------|
| 19.24.1 SM.ACC | 686 |
| 19.24.2 SM.DEC | 687 |
| 19.24.3 SM.I1 | 688 |
| 19.24.4 SM.I2 | 689 |
| 19.24.5 SM.MODE | 690 |
| 19.24.6 SM.MOVE | 693 |
| 19.24.7 SM.T1 | 694 |
| 19.24.8 SM.T2 | 695 |
| 19.24.9 SM.V1 | 696 |
| 19.24.10 SM.V2 | 697 |

19.24.1 SM.ACC

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the acceleration ramp for service motion if set to greater than 0. |
| Units | Depends on UNIT.ACCROTARY or UNIT.ACCLINEAR Rotary: rps/s, rpm/s, deg/s ² , (custom units)/s ² , rad/s ² Linear: counts/s ² , mm/s ² , μm/s ² , (custom units)/s ² |
| Range | Note: The range and default values of (custom units)/s ² units depend on the values of PIN and POUT. The range and default values listed in this table are derived from the default values of PIN and POUT. Rotary: 0.0 to 833,333.333 rps/s 0.0 to 50,000,000.000 rpm/s 0.0 to 300,000,000.000 deg/s ² 0.0 to 4,166,666.752 (custom units)/s ² 0.0 to 5,235,987.968 rad/s ² Linear: 0 to 3,579,139,408,000.000 counts/s ² 0 to 833,333.333*MOTOR.PITCH mm/s ² 0 to 2,147,483.647*MOTOR.PITCH μm/s ² 0 to 2,147,483.647 (custom units)/s ² |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-13-06-000 |

Description

Sets the acceleration ramp for service motion if set to greater than 0.

Related Topics

Service Motion (→ p. 89)

19.24.2 SM.DEC

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the deceleration ramp for service motion if set to greater than 0. |
| Units | Depends on UNIT.ACCROTARY or UNIT.ACCLINEAR Rotary: rps/s, rpm/s, deg/s ² , (custom units)/s ² , rad/s ² Linear: counts/s ² , mm/s ² , μm/s ² , (custom units)/s ² |
| Range | Note: The range and default values of (custom units)/s ² units depend on the values of PIN and POUT. The range and default values listed in this table are derived from the default values of PIN and POUT. Rotary: 0.0 to 833,333.333 rps/s 0.0 to 50,000,000.000 rpm/s 0.0 to 300,000,000.000 deg/s ² 0.0 to 4,166,666.752 (custom units)/s ² 0.0 to 5,235,987.968 rad/s ² Linear: 0 to 3,579,139,408,000.000 counts/s ² 0 to 833,333.333*MOTOR.PITCH mm/s ² 0 to 833,333,333.333*MOTOR.PITCH μm/s ² 0 to 4,166,666.667 (custom units)/s ² |
| Default Value | 0 |
| Data Type | Float |
| Start Version | M_01-13-06-000 |

Description

Sets the deceleration ramp for service motion if set to greater than 0.

Related Topics

Service Motion (→ p. 89)

19.24.3 SM.I1

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets service motion current 1; active in opmode 0 (torque) only. |
| Units | A |
| Range | –Drive peak current to +Drive peak current |
| Default Value | 0.025 · Drive peak current |
| Data Type | Float |
| See Also | SM.ACCTYPE, SM.I2, SM.MODE, SM.MOVE, SM.T1, SM.T2, SM.V1, SM.V2, SM.VPM1, SM.VPM2 |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5175h/0 | M_01-00-00-000 |

Description

SM.I1 defines the current that is used in service motion modes 0 and 1 (see SM.MODE (→ p. 690)).

Related Topics

Service Motion (→ p. 89)

19.24.4 SM.I2

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets service motion current 2; active in opmode 0 (torque) only. |
| Units | A |
| Range | –Drive peak current to +Drive peak current |
| Default Value | 0.025 · Drive peak current |
| Data Type | Float |
| See Also | SM.ACCTYPE, SM.I1, SM.MODE, SM.MOVE, SM.T1, SM.T2, SM.V1, SM.V2, SM.VPM1, SM.VPM2 |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5176h/0 | M_01-00-00-000 |

Description

SM.I2 defines the current that is used in service motion mode 1 (see SM.MODE (→ p. 690)).

Related Topics

Service Motion (→ p. 89)

19.24.5 SM.MODE

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the service motion mode. |
| Units | N/A |
| Range | 0 to 2 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | SM.I1 (→ p. 688), SM.I2 (→ p. 689), SM.MOVE (→ p. 693), SM.T1 (→ p. 694) SM.T2 (→ p. 695), SM.V1 (→ p. 696), SM.V2 (→ p. 697), DRV.ACC (→ p. 366) DRV.DEC (→ p. 379) |
| Start Version | M_01-01-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5177h/0 | M_01-01-00-000 |

Description

SM.MODE defines the mode of service motion for each loop. Two types of service motion are available :

- A constant motion in one direction (endless or for a certain amount of time).
- An alternating motion.

The possible modes for this parameter are described in the following table:

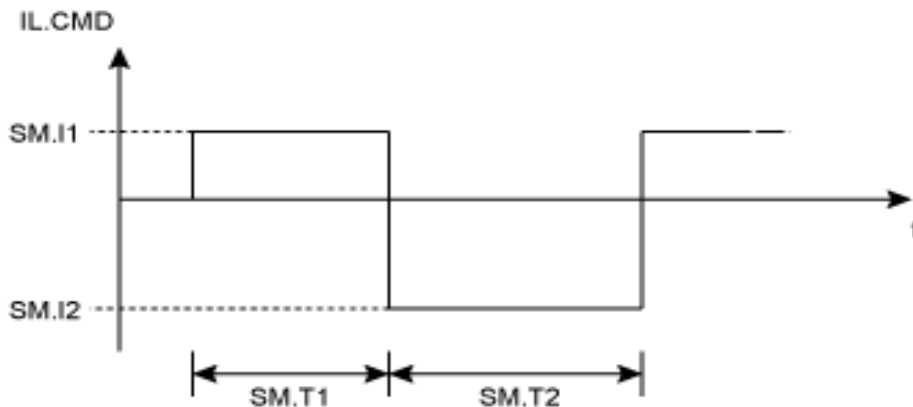
| SM.MODE | Description | Requirements |
|---------|--|---|
| 0 | <p>Constant motion in closed current loop mode of operation.</p> <ul style="list-style-type: none"> • DRV.OPMODE 0: The drive generates a constant current command value (SM.I1) for a certain amount of time (if SM.T1>0) or endless (if SM.T1=0). The drive will not generate any ramps in this mode of operation. • DRV.OPMODE 1 or 2: The drive generates a constant velocity command value (SM.V1) for a certain amount of time (if SM.T1>0) or endless (if SM.T1=0). The drive generates acceleration and deceleration ramps according to the DRV.ACC and DRV.DEC setting in this mode of operation. <p>The service motion can be stopped by using the DRV.STOP command.</p> | <p>DRV.OPMODE = 0,1, or 2 DRV.CMDSOURCE = 0</p> |

| SM.MODE | Description | Requirements |
|---------|---|--|
| 1 | <ul style="list-style-type: none"> • DRV.OPMODE 0: The drive generates a current command value (SM.I1) for a certain amount of time (SM.T1). Afterwards the drive generates a current command value (SM.I2) for another certain amount of time (SM.T2). This sequence is repeated as long as a DRV.STOP command occurs. The drive will not generate any ramps in this mode of operation. • DRV.OPMODE 1 or 2: The drive generates a velocity command value (SM.V1) for a certain amount of time (SM.T1). Afterwards the drive generates a velocity command value (SM.V2) for another certain amount of time (SM.T2). This sequence is repeated as long as a DRV.STOP command occurs. The drive will generate an acceleration and deceleration ramps according to the DRV.ACC and DRV.DEC setting in this mode of operation. | DRV.OPMODE = 0, 1, or 2 DRV.CMDSOURCE = 0 |
| 2 | This mode executes the same service motion as mode 0. However, the motion is described by SM.I2, SM.T2 and SM.V2. This enables a change on the fly and is mostly used under fieldbus control. | DRV.OPMODE = 0, 1, or 2 DRV.CMDSOURCE = 0 |

Ramps

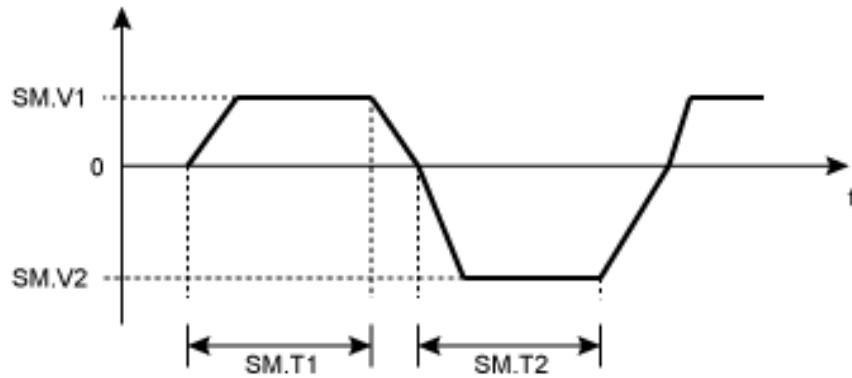
The drive uses DRV.ACC and DRV.DEC for the ramps in DRV.OPMODE 1 (closed velocity) and 2 (closed position). The drive does not generate any ramps in service motion mode 0 and 1.

Service Motion for DRV.OPMODE 0 and SM.MODE 1



Service motion for DRV.OPMODE 1 or 2 and SM.MODE 1

DRV.OPMODE 1: VL.CMD
 DRV.OPMODE 2: d/dt (PL.CMD)



The deceleration process from SM.V1 or SM.V2 to 0 is not included in SM.T1 and SM.T2, respectively. SM.T1 and SM.T2 start as soon as the command value has reached the velocity 0.

Related Topics

Service Motion (→ p. 89)

19.24.6 SM.MOVE

| General Information | |
|---------------------|----------------------------|
| Type | Command |
| Description | Starts the service motion. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| See Also | SM.MODE |
| Start Version | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

This command starts the service motion that has been selected by the SM.MODE parameter.

Related Topics

Service Motion (→ p. 89)

19.24.7 SM.T1

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the service motion time 1. |
| Units | ms |
| Range | 0 to 65,535 ms |
| Default Value | 500 ms |
| Data Type | Integer |
| See Also | SM.I1, SM.I2, SM.MODE, SM.MOVE, SM.T2, SM.V1, SM.V2, SM.VPM1, SM.VPM2 |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5179h/0 | M_01-00-00-000 |

Description

SM.T1 defines the time of the service motion that is used in all service motion modes (see SM.MODE). For an alternating service motion mode, SM.T1 may not be set to 0.

Related Topics

Service Motion (→ p. 89)

19.24.8 SM.T2

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the service motion time 2. |
| Units | ms |
| Range | 0 to 65,535 ms |
| Default Value | 500 ms |
| Data Type | Integer |
| See Also | SM.I1, SM.I2, SM.MODE, SM.MOVE, SM.T1, SM.V1, SM.V2, SM.VPM1, SM.VPM2 |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 517Ah/0 | M_01-00-00-000 |

Description

SM.T2 defines the time of the service motion that is used in service motion modes 1, 3, and 5 (see SM.MODE).

Related Topics

Service Motion (→ p. 89)

19.24.9 SM.V1

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets service motion velocity 1; active in opmode 1 (velocity) and 2 (position). |
| Units | Depends on UNIT.ACCROTARY (pg 1) or UNIT.ACCLINEAR Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: Counts/s, mm/s, $\mu\text{m/s}$, custom units/s |
| Range | Rotary: -15,000.000 to 15,000.000 rpm -250.000 to 250.000 rps -90,000.000 to 90,000.000 deg/s -1,250.000 to 1,250.000 custom units/s -1,570.796 to 1,570.796 rad/s Linear: -1,073,741,824,000.000 to 1,073,741,824,000.000 counts/s -250.000*MOTOR.PITCH (\rightarrow p. 617) to 250.000*MOTOR.PITCH (\rightarrow p. 617) mm/s -250,000.000*MOTOR.PITCH (\rightarrow p. 617) to 250,000.000*MOTOR.PITCH (\rightarrow p. 617) $\mu\text{m/s}$ -1,250.000 to 1,250.000 custom units/s |
| Default Value | Rotary: 60.000 rpm 1.000 rps 359.999 deg/s 5.000 custom units/s 6.283 rad/s Linear: 0.001 Counts/s 1.000*MOTOR.PITCH (\rightarrow p. 617) mm/s 999.998*MOTOR.PITCH (\rightarrow p. 617) $\mu\text{m/sec}$ 5.000 custom units/s |
| Data Type | Float |
| See Also | SM.I1, SM.I2, SM.MODE, SM.MOVE, SM.T1, SM.T2, SM.V2 |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3408h/1 | M_01-00-00-000 |

Description

SM.V1 defines the velocity that is used in service motion modes 0 and 1 (see SM.MODE (\rightarrow p. 690)) in the closed velocity and position mode of operation.

Related Topics

11.1 Service Motion

19.24.10 SM.V2

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets service motion velocity 2; active in opmode 1 (velocity) and 2 (position). |
| Units | Depends on UNIT.ACROTARY (pg 1) or UNIT.ACCLINEAR Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: counts/s, mm/s, μ m/s, custom units/s |
| Range | Rotary: -15,000.000 to 15,000.000 rpm -250.000 to 250.000 rps -90,000.000 to 90,000.000 deg/s -1,250.000 to 1,250.000 custom units/s -1,570.796 to 1,570.796 rad/s Linear: -1,073,741,824,000.000 to 1,073,741,824,000.000 counts/s -250.000*MOTOR.PITCH (→ p. 617) to 250.000*MOTOR.PITCH (→ p. 617) mm/s -250,000.000*MOTOR.PITCH (→ p. 617) to 250,000.000*MOTOR.PITCH (→ p. 617) μ m/s -1,250.000 to 1,250.000 custom units/s |
| Default Value | Rotary: -60.000 rpm -1.000 rps -359.999 deg/s -5.000 custom units/s -6.283 rad/s Linear: -0.001 counts/s -1.000*MOTOR.PITCH (→ p. 617) mm/s -999.998*MOTOR.PITCH (→ p. 617) μ m/sec -5.000 custom units/s |
| Data Type | Float |
| See Also | SM.I1, SM.I2, SM.MODE, SM.MOVE, SM.T1, SM.T2, SM.V1 |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3408h/1 | M_01-00-00-000 |

Description

SM.V2 defines the velocity that is used in service motion mode 1 (see SM.MODE) in the closed velocity and position mode of operation.

Related Topics

11.1 Service Motion

19.25 STO Parameters

This section describes the STO parameters.

| | |
|--------------------------------|------------|
| 19.25.1 STO.STATE | 699 |
|--------------------------------|------------|

19.25.1 STO.STATE

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Returns the status of the safe torque off. |
| Units | N/A |
| Range | 0 to 1 for drives with only one STO (AKD-M003 up to AKD-M024) 0 to 3 for drives that support STO2 (AKD-M048) |
| Default Value | N/A |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

STO.STATE returns the status of the safe torque off.

AKD-M003 up to AKD-M024 (one STO channel)

| Value | Description |
|-------|---|
| 1 | Safe torque on (no STO fault/warning) |
| 0 | Safe torque off (STO fault and warning are present) |

1 - Safe torque on (no safe torque off fault/warning).

0 - Safe torque off (safe torque off fault/warning).

When STO.STATE = 1, the drive can be enabled.

AKD-M048 (two STO channels)

An STO fault and warning are generated when either STO1 or STO2 are not on. Use STO.STATE to determine the source (STO1 or STO2) of the STO fault. EWV¹ also displays the status of both STO1 and STO2 at the bottom of the screen.

| Value | Description |
|-------|---|
| 3 | STO1 and STO2 are on (no STO fault/warning) |
| 2 | STO1 is off, and STO2 is on (STO fault and warning are present) |
| 1 | STO1 is on, and STO2 is off (STO fault and warning are present) |
| 0 | STO1 and STO2 are off (STO fault and warning are present) |

Related Topics

Safe Torque Off (STO) (→ p. 1)

10.1 Limits

¹Embedded Workbench Views

19.26 SWLS Parameters

This section describes the SWLS parameters.

| | |
|----------------------------------|------------|
| 19.26.1 SWLS.EN | 701 |
| 19.26.2 SWLS.LIMIT0 | 702 |
| 19.26.3 SWLS.LIMIT1 | 703 |
| 19.26.4 SWLS.STATE | 704 |

19.26.1 SWLS.EN

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Enables and disables software travel limit switches. |
| Units | N/A |
| Range | 0 to 3 |
| Default Value | 0 |
| Data Type | U8 |
| See Also | DRV.MOTIONSTAT |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 517Eh/0 | M_01-00-00-000 |

Description

This parameter enables the software travel limit switches. The software limit switches are only active if the axis is homed.

| Value | Description |
|-------|--|
| 0 | Disables both limit switches. |
| 1 | Enables limit switch 0. |
| 2 | Enables limit switch 1. |
| 3 | Enables limit switch 0 and limit switch 1. |

If SWLS.EN = 3, then whichever limit switch is set to a greater value will be used as the upper limit switch. For all other values of SWLS.EN, use SWLS.LIMIT0 for the lower limit switch, and SWLS.LIMIT1 for the upper limit switch.

Example

Bit 0 = 0: Disable SWLS.LIMIT0

Bit 0 = 1: Enable SWLS.LIMIT0

Bit 1 = 0: Disable SWLS.LIMIT1

Bit 1 = 1: Enable SWLS.LIMIT1

Related Topics

10.1 Limits

19.26.2 SWLS.LIMIT0

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the position of the software travel limit switch 0. |
| Units | Position units |
| Range | -9,007,199,254,740,992 to 9,007,199,254,740,991 |
| Default Value | 0 |
| Data Type | S64 |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 607Dh/1 | M_01-00-00-000 |

Description

This parameter sets the compare register for the software limit switch 0. This value can be either the lower or the upper software limit switch register, depending on the configuration of the software limit switches:

- If only one limit switch is enabled (SWLS.EN = 1 or 2), then SWLS.LIMIT0 is the lower limit, and SWLS.LIMIT1 is the upper limit.
- If both limit switches are enabled (SWLS.EN = 3), then whichever switch is set to a larger value is the upper limit switch. The other switch becomes the lower limit switch.

These switches can be used in addition to hardware limit switches. The software limit switches are only active if the axis is homed. For more information about homing, please refer to the HOME Parameters and DRV.MOTIONSTAT.

Related Topics

10.1 Limits

19.26.3 SWLS.LIMIT1

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the position of the software travel limit switch 0. |
| Units | Position units |
| Range | -9,007,199,254,740,992 to 9,007,199,254,740,991 |
| Default Value | 1,048,576.000 counts, 16-bit (firmware versions M_01-02-00-000 and above) 68,719,476,736 counts (for firmware version M_01-01-00-000) |
| Data Type | S64 |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 607Dh/2 | M_01-00-00-000 |

Description

This parameter sets the compare register for the software limit switch 1. This value can be either the lower or the upper software limit switch register, depending on the configuration of the software limit switches:

- If only one limit switch is enabled (SWLS.EN = 1 or 2), then SWLS.LIMIT0 is the lower limit, and SWLS.LIMIT1 is the upper limit.
- If both limit switches are enabled (SWLS.EN = 3), then whichever switch is set to a larger value is the upper limit switch. The other switch becomes the lower limit switch.

These switches can be used in addition to hardware limit switches. The software limit switches are only active if the axis is homed. For more information about homing, please refer to the HOME Parameters and DRV.MOTIONSTAT.

Related Topics

10.1 Limits

19.26.4 SWLS.STATE

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the actual status of software limit switches. |
| Units | N/A |
| Range | 0 to 3 |
| Default Value | 0 |
| Data Type | U8 |
| Start Version | M_01-00-00-000 |

Description

This parameter reads the status word of the software limit switches. The status word indicates the result of the compare between the software limit switch compare register and the actual position of the position loop.

Example

Bit 0 = 0: SWLS.LIMIT0 (→ p. 702) is not active.

Bit 0 = 1: SWLS.LIMIT0 is active.

Bit 1 = 0: SWLS.LIMIT1 (→ p. 703) is not active.

Bit 1 = 1: SWLS.LIMIT1 is active.

Bits 2 to 7 are currently not in use.

Related Topics

10.1 Limits

19.27 TEMP Parameters

This section describes the TEMP parameters.

| | |
|---|------------|
| 19.27.1 TEMP.CONTROL | 706 |
| 19.27.2 TEMP.POWER1 to TEMP.POWER3 | 707 |

19.27.1 TEMP.CONTROL

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the temperature of the control board temperature sensor. |
| Units | Degrees Celsius |
| Range | -127 to 128 |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-09-00-000 |

Description

Reads the temperature of the control board temperature sensor.

Related Topics

DRV.TEMPERATURES (→ p. 439) | TEMP.POWER1 to TEMP.POWER3 (→ p. 707)

19.27.2 TEMP.POWER1 to TEMP.POWER3

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the temperature of the power board(s) temperature sensor (s). |
| Units | Degrees Celsius |
| Range | -127 to 128 |
| Default Value | N/A |
| Data Type | Integer |
| Start Version | M_01-09-00-000 |

Description

Reads the temperature of the power board(s) temperature sensor(s). Most AKD PDMMs have only one power board sensor which is read by TEMP.POWER1.

Related Topics

DRV.TEMPERATURES (→ p. 439) | TEMP.CONTROL (→ p. 706)

19.28 USER Parameters

This section describes the USER parameters.

| | |
|--|------------|
| 19.28.1 USER.INT1 to USER.INT24 | 709 |
|--|------------|

19.28.1 USER.INT1 to USER.INT24

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Stores an integer value for future reference. |
| Units | N/A |
| Range | -2,147,483,648 to 2,147,483,647 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

Description

This parameter stores an integer value. To retain this value after a power-cycle, a "Save to Device" command must be performed from the Parameter Load/Save view in EWV¹.

¹Embedded Workbench Views

19.29 VBUS Parameters

This section describes the VBUS parameters.

| | |
|-------------------------------------|------------|
| 19.29.1 VBUS.HALFVOLT | 711 |
| 19.29.2 VBUS.OVFTHRESH | 712 |
| 19.29.3 VBUS.OVWTHRESH | 713 |
| 19.29.4 VBUS.RMSLIMIT | 714 |
| 19.29.5 VBUS.UVFTHRESH | 715 |
| 19.29.6 VBUS.UVMODE | 716 |
| 19.29.7 VBUS.UVWTHRESH | 717 |
| 19.29.8 VBUS.VALUE | 718 |

19.29.1 VBUS.HALFVOLT

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Changes voltage thresholds for -07 drives. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-04-01-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5261h/0 | M_01-04-01-000 |

Description

This parameter is used to change voltage thresholds to allow -07 drive types to work at an -06 drive type's voltage thresholds. For information on voltage ratings and drive types see AKD PDMM Models (→ p. 26).

When the VBUS.HALFVOLT value is changed, the revised thresholds take effect immediately.

VBUS.HALFVOLT only affects -07 drives.

| Value | Description |
|-------|---|
| 0 | The original voltage thresholds are used for the functions mentioned below. |
| 1 | -07 drives use voltage thresholds of -06 drives. |

This parameter has an impact on the following voltage-thresholds:

1. DC-bus over-voltage threshold (see VBUS.OVFTHRESH (→ p. 712)).
2. The regen-resistor enable/disable voltage thresholds.
3. The inrush-relay enable/disable voltage thresholds.

NOTE

The DC-bus under voltage fault threshold (see VBUS.UVFTHRESH (→ p. 715)) is a user selectable command. It means that the user is responsible for setting the under voltage threshold to a proper value in case that the AKD PDMM is supplied with a lower DC-bus voltage than the rated voltage.

19.29.2 VBUS.OVFTHRESH

| General Information | |
|---------------------|-------------------------------------|
| Type | R/O Parameter |
| Description | Reads the over voltage fault level. |
| Units | VDC |
| Range | 0 to 900 VDC |
| Default Value | N/A |
| Data Type | Integer |
| See Also | VBUS.UVFTHRESH |
| Start Version | M_01-00-00-000 |

Description

VBUS.OVFTHRESH reads the over voltage fault level for the DC bus.

This value is read from the drive EEPROM and varies according to the drive type.

Related Topics

Regeneration (→ p. 43)

19.29.3 VBUS.OVWTHRESH

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets voltage level for over voltage warning. |
| Units | VDC |
| Range | 0 to 900 VDC |
| Default Value | 0 VDC (warning disabled) |
| Data Type | U16 |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 518Eh/0 | M_01-00-00-000 |

Description

If VBUS.VALUE value exceeds VBUS.OVWTHRESH, then a warning is generated. The warning is removed if the voltage returns to below 97% (1/32) of the overvoltage warning level.

Related Topics

Regeneration (→ p. 43)

19.29.4 VBUS.RMSLIMIT

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the limit for the bus capacitors load. |
| Units | Vrms |
| Range | N/A |
| Default Value | N/A |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

Description

This parameter reads the limit of the bus capacitor load. When the bus capacitor loads exceeds this limit, the drive generates fault F503.

Excessive bus capacitor load may indicate a disconnected main supply phase.

Related Topics

Regeneration (→ p. 43)

19.29.5 VBUS.UVFTHRESH

| General Information | |
|---------------------|-------------------------------------|
| Type | R/W Parameter |
| Description | Sets the under voltage fault level. |
| Units | VDC |
| Range | 90 to 420 VDC |
| Default Value | 90 VDC |
| Data Type | Integer |
| See Also | VBUS.OVFTHRESH |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 361Dh/0 | M_01-00-00-000 |

Description

VBUS.UVFTHRESH sets the undervoltage fault level of the DC bus.

The default value is read from the EEPROM, but can be modified by the user and stored on the NV RAM. This value varies according to drive type.

Related Topics

Regeneration (→ p. 43)

19.29.6 VBUS.UVMODE

| General Information | |
|---------------------|-----------------------------------|
| Type | N/V Parameter |
| Description | Indicates undervoltage (UV) mode. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 1 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3617h/0 | M_01-00-00-000 |

Description

This parameter indicates undervoltage (UV) mode.

When VBUS.UVMODE = 0, an undervoltage fault is issued whenever the DC bus goes below the undervoltage threshold.

When VBUS.UVMODE = 1, an undervoltage fault is issued whenever the DC bus goes below the undervoltage threshold and the controller attempts to enable the drive (software or hardware enable).

Related Topics

Regeneration (→ p. 43)

19.29.7 VBUS.UVWTHRESH

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets voltage level for undervoltage warning. |
| Units | VDC |
| Range | 0 to 900 VDC |
| Default Value | 10 volts above the default value of the under voltage fault threshold (VBUS.UVFTHRESH). The default value of VBUS.UVFTHRESH is hardware dependent. |
| Data Type | U16 |
| See Also | VBUS.UVFTHRESH (→ p. 715) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3495h/0 | M_01-00-00-000 |

Description

If VBUS.VALUE value drops below VBUS.UVWTHRESH, then a warning is generated.

Related Topics

Regeneration (→ p. 43)

19.29.8 VBUS.VALUE

| General Information | |
|---------------------|-----------------------|
| Type | R/O Parameter |
| Description | Reads DC bus voltage. |
| Units | VDC |
| Range | 0 to 900 VDC |
| Default Value | N/A |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 361Ah/0 | M_01-00-00-000 |

Description

VBUS.VALUE reads the DC bus voltage.

Related Topics

6.4 Regeneration

19.30 VL Parameters

This section describes the VL parameters.

| | |
|--|-----|
| 19.30.1 VL.ARPF1 TO VL.ARPF4 | 720 |
| 19.30.2 VL.ARPQ1 TO VL.ARPQ4 | 722 |
| 19.30.3 VL.ARTYPE1 TO VL.ARTYPE4 | 724 |
| 19.30.4 VL.ARZF1 TO VL.ARZF4 | 725 |
| 19.30.5 VL.ARZQ1 TO VL.ARZQ4 | 726 |
| 19.30.6 VL.BUSFF | 728 |
| 19.30.7 VL.CMD | 729 |
| 19.30.8 VL.CMDU | 730 |
| 19.30.9 VL.ERR | 731 |
| 19.30.10 VL.FB | 732 |
| 19.30.11 VL.FBFILTER | 733 |
| 19.30.12 VL.FBSOURCE | 734 |
| 19.30.13 VL.FBUNFILTERED | 735 |
| 19.30.14 VL.FF | 736 |
| 19.30.15 VL.FFDELAY | 737 |
| 19.30.16 VL.GENMODE | 738 |
| 19.30.17 VL.KBUSFF | 739 |
| 19.30.18 VL.KI | 740 |
| 19.30.19 VL.KIMODE | 741 |
| 19.30.20 VL.KO | 742 |
| 19.30.21 VL.KP | 743 |
| 19.30.22 VL.KVFF | 745 |
| 19.30.23 VL.LIMITN | 746 |
| 19.30.24 VL.LIMITP | 748 |
| 19.30.25 VL.LMJR | 749 |
| 19.30.26 VL.MODEL | 750 |
| 19.30.27 VL.OBSBW | 751 |
| 19.30.28 VL.OBSMODE | 752 |
| 19.30.29 VL.THRESH | 753 |
| 19.30.30 VL.VFTHRESH | 755 |

19.30.1 VL.ARPF1 TO VL.ARPF4

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the natural frequency of the pole (denominator) of anti-resonance (AR) filters 1, 2, 3, and 4; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Hz |
| Range | 5 to 5,000 Hz |
| Default Value | 500 Hz |
| Data Type | Float |
| See Also | VL.ARPQ1 TO VL.ARPQ4 (→ p. 722), VL.ARZF1 TO VL.ARZF4 (→ p. 725), Sets the Q of the zero (numerator) of anti-resonance filter #1; active in opmodes 1 (velocity) and 2 (position) only. (→ p. 726) |
| Start Version | M_01-02-00-000 |

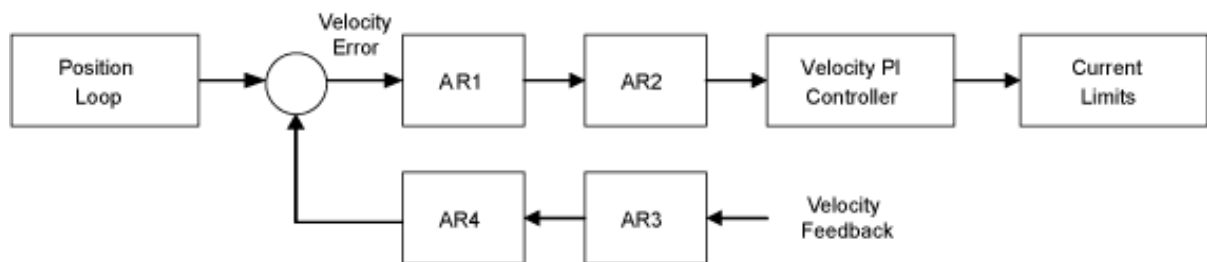
| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3406h/1 | VL.ARPF1 |
| | 3406h/2 | VL.ARPF2 |
| | 3406h/3 | VL.ARPF3 |
| | 3406h/4 | VL.ARPF4 |

Description

VL.ARPF1 sets the natural frequency of the pole (denominator) of AR filter 1. This value is F_P in the approximate transfer function of the filter:

$$ARx(s) = [s^2 / (2\pi F_Z)^2 + s / (Q_Z 2\pi F_Z) + 1] / [s^2 / (2\pi F_P)^2 + s / (Q_P 2\pi F_P) + 1]$$

The following block diagram describes the AR filter function; note that AR1 and AR2 are in the forward path, while AR3 and AR4 are applied to feedback:



AR1, AR2, AR3, and AR4 are used in velocity and position mode, but are disabled in torque mode.

Discrete time transfer function (applies to all AR filters)

The velocity loop compensation is actually implemented as a digital discrete time system function on the DSP. The continuous time transfer function is converted to the discrete time domain by a backward Euler mapping:

$$s \approx (1-z^{-1})/t, \text{ where } t = 62.5 \mu\text{s}$$

The poles are prewarped to F_P and the zeros are prewarped to F_Z .

Related Topics

- 1 Velocity Loop

19.30.2 VL.ARPQ1 TO VL.ARPQ4

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the Q of the pole (denominator) of anti-resonance (AR) filter 1; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | None |
| Range | 0.2 to 20 |
| Default Value | 0.5 |
| Data Type | Float |
| See Also | VL.ARPF1 TO VL.ARPF4 (→ p. 720), VL.ARZF1 TO VL.ARZF4 (→ p. 725), VL.ARZQ1 TO VL.ARZQ4 (→ p. 726) |
| Start Version | M_01-02-00-000 |

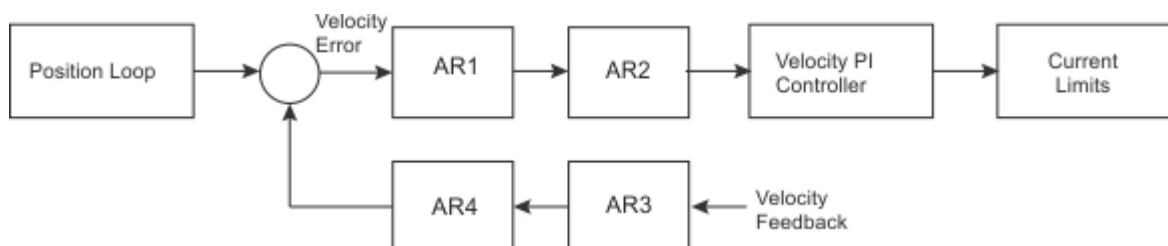
| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3406h/5 | VL.ARPQ1 |
| | 3406h/6 | VL.ARPQ2 |
| | 3406h/7 | VL.ARPQ3 |
| | 3406h/8 | VL.ARPQ4 |

Description

VL.ARPQ1 sets the Q (quality factor) of the pole (denominator) of AR filter 1. This value is Q_P in the approximate transfer function of the filter:

$$AR_x(s) = [s^2 / (2\pi F_Z)^2 + s / (Q_Z 2\pi F_Z) + 1] / [s^2 / (2\pi F_P)^2 + s / (Q_P 2\pi F_P) + 1]$$

The following block diagram describes the AR filter function; note that AR1 and AR2 are in the forward path, while AR3 and AR4 are applied to feedback:



AR1, AR2, AR3, and AR4 are used in velocity and position mode, but are disabled in torque mode.

Discrete time transfer function (applies to all AR filters)

The velocity loop compensation is actually implemented as a digital discrete time system function on the DSP. The continuous time transfer function is converted to the discrete time domain by a backward Euler mapping:

$$s \approx (1-z^{-1})/t, \text{ where } t = 62.5 \mu\text{s}$$

The poles are prewarped to F_P and the zeros are prewarped to F_Z .

Related Topics

- 1 Velocity Loop

19.30.3 VL.ARTYPE1 TO VL.ARTYPE4

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Indicates the method used to calculate BiQuad coefficients; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | N/A |
| Range | 0 |
| Default Value | 0 |
| Data Type | U8 |
| See Also | N/A |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3405h/1 | VL.ARTYPE1 |
| | 3405h/2 | VL.ARTYPE2 |
| | 3405h/3 | VL.ARTYPE3 |
| | 3405h/4 | VL.ARTYPE4 |

Description

These parameters indicate the method used to calculate the biquad coefficients VL.ARPFx, VL.ARPQx, VL.ARZFx, and VL.ARZQx. A value of 0 indicates that the coefficients are set directly. This parameter has no effect on the filter itself, but is only used to determine the original design parameters. Currently, only the value of 0 is supported.

Related Topics

- 1 Velocity Loop

19.30.4 VL.ARZF1 TO VL.ARZF4

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the natural frequency of the zero (numerator) of anti-resonance (AR)filter 1; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Hz |
| Range | 5 to 5,000 Hz |
| Default Value | 500 Hz |
| Data Type | Float |
| See Also | VL.ARPF1 TO VL.ARPF4 (→ p. 720), VL.ARPQ1 TO VL.ARPQ4 (→ p. 722), VL.ARZQ1 TO VL.ARZQ4 (→ p. 726) |
| Start Version | M_01-02-00-000 |

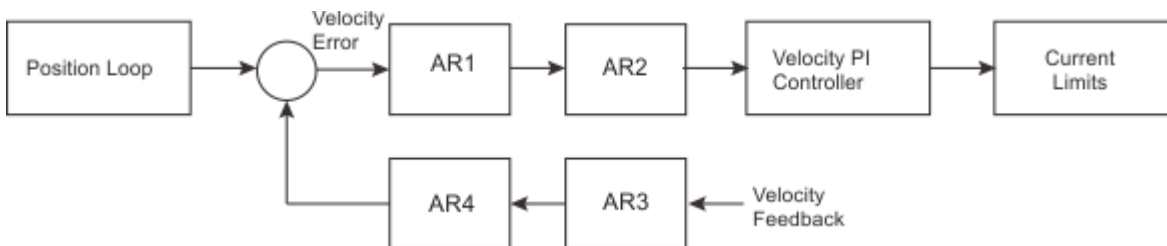
| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3406h/9 | VL.ARZF1 |
| | 3406h/A | VL.ARZF2 |
| | 3406h/B | VL.ARZF3 |
| | 3406h/C | VL.ARZF4 |

Description

VL.ARZF1 sets the natural frequency of the zero (numerator) of AR filter 1. This value is F_Z in the approximate transfer function of the filter:

$$ARx(s) = [s^2 / (2\pi F_Z)^2 + s / (Q_Z 2\pi F_Z) + 1] / [s^2 / (2\pi F_P)^2 + s / (Q_P 2\pi F_P) + 1]$$

The following block diagram describes the AR filter function; note that AR1 and AR2 are in the forward path, while AR3 and AR4 are applied to feedback:



AR1, AR2, AR3, and AR4 are used in velocity and position mode, but are disabled in torque mode.

Discrete time transfer function (applies to all AR filters)

The velocity loop compensation is actually implemented as a digital discrete time system function on the DSP. The continuous time transfer function is converted to the discrete time domain by a backward Euler mapping:

$$s \approx (1-z^{-1})/t, \text{ where } t = 62.5 \mu\text{s}$$

The poles are prewarped to F_P and the zeros are prewarped to F_Z .

Related Topics

- 1 Velocity Loop

19.30.5 VL.ARZQ1 TO VL.ARZQ4

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the Q of the zero (numerator) of anti-resonance filter #1; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | N/A |
| Range | 0.1 to 5 |
| Default Value | 0.5 |
| Data Type | Float |
| See Also | VL.ARPF1 TO VL.ARPF4 (→ p. 720), VL.ARPQ1 TO VL.ARPQ4 (→ p. 722), VL.ARZF1 TO VL.ARZF4 (→ p. 725) |
| Start Version | M_01-02-00-000 |

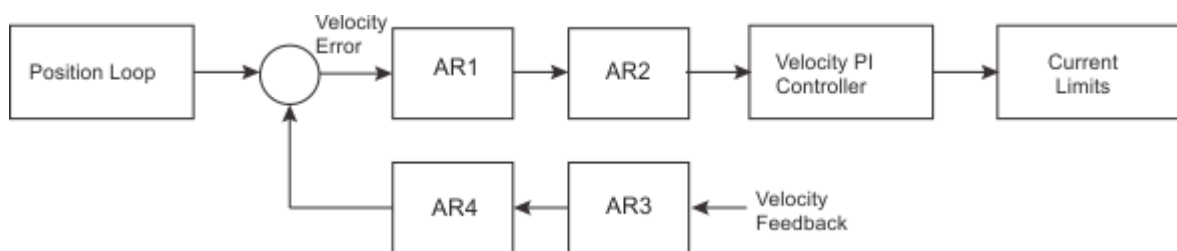
| Fieldbus | Index/Subindex | | Object Start Version |
|--------------------------|----------------|----------|----------------------|
| EtherCAT COE and CANopen | 3406h/D | VL.ARZQ1 | M_01-02-00-000 |
| | 3406h/E | VL.ARZQ2 | |
| | 3406h/F | VL.ARZQ3 | |
| | 3406h/10 | VL.ARZQ4 | |

Description

VL.ARZQ1 sets the Q (quality factor) of the zero (numerator) of AR filter 1. This value is Q_z in the approximate transfer function of the filter:

$$AR1(s) = [s^2 / (2\pi F_z)^2 + s / (Q_z 2\pi F_z) + 1] / [s^2 / (2\pi F_p)^2 + s / (Q_p 2\pi F_p) + 1]$$

The following block diagram describes the AR filter function; note that AR1 and AR2 are in the forward path, while AR3 and AR4 are applied to feedback:



AR1, AR2, AR3 and AR4 are used in velocity and position mode, but are disabled in torque mode.

Discrete time transfer function (applies to all AR filters)

The velocity loop compensation is actually implemented as a digital discrete time system function on the DSP. The continuous time transfer function is converted to the discrete time domain by a backward Euler mapping:

$$s \approx (1-z^{-1})/t, \text{ where } t = 62.5 \mu\text{s}.$$

The poles are prewarped to F_p and the zeros are prewarped to F_z .

Related Topics

1 Velocity Loop

19.30.6 VL.BUSFF

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Displays the velocity loop feedforward value injected by the field-bus; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Depends on UNIT.VROTARY (pg 1) or UNIT.VLINEAR (pg 1) Rotary: rpm, rps, deg/s, (custom units)/s, rad/s Linear: counts/s, mm/s, $\mu\text{m/s}$, (custom units)/s |
| Range | 0.0 to VL.LIMITP (→ p. 748) |
| Default Value | 0.0 |
| Data Type | Float |
| See Also | VL.FF (→ p. 736), VL.KBUSFF (→ p. 739) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 60B1h/0 | M_01-00-00-000 |

Description

This parameter displays the velocity loop feedforward value injected by the fieldbus.

Related Topics

- 1 Velocity Loop

19.30.7 VL.CMD

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the actual velocity command; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Depends on UNIT.VROTARY (pg 1) or UNIT.VLINEAR (pg 1) Rotary: rpm, rps, deg/s, (custom units)/s, rad/s Linear: counts/s, mm/s, µm/s, (custom units)/s |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| See Also | VL.FB (→ p. 732), VL.CMDU (→ p. 730), VL.LIMITP (→ p. 748), VL.LIMITN (→ p. 746) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 606Bh/0 | M_01-00-00-000 |

Description

VL.CMD returns the actual velocity command as it is received in the velocity loop entry after all velocity limits (such as VL.LIMITN (→ p. 746) and VL.LIMITP (→ p. 748)). See velocity loop design diagram for more details.

Related Topics

- 1 Velocity Loop
- 1 Position Loop

19.30.8 VL.CMDU

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the user velocity command; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Depends on UNIT.VROTARY (pg 1) or UNIT.VLINEAR (pg 1) Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: counts/s, mm/s, $\mu\text{m/s}$, custom units/s |
| Range | Rotary -15,000.000 to 15,000.000 rpm -250.000 to 250.000 rps -90000.000 to 90000.000 deg/s -1250.000 to 1250.000 custom units/s -1570.796 to 1570.796 rad/s Linear -1,073,741,824,000.000 to 1,073,741,824,000.000 counts/s -8,000.000 to 8,000.000 mm/s -8,000,000.000 to 8,000,000.000 $\mu\text{m/s}$ -1,250.000 to 1,250.000 custom units/s |
| Default Value | 0 |
| Data Type | Float |
| See Also | VL.FB (→ p. 732), VL.CMD (→ p. 729), DRV.OPMODE (→ p. 430), DRV.CMDSOURCE (→ p. 375), VL.LIMITN (→ p. 746), VL.LIMITP (→ p. 748) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 60FFh/0 | M_01-00-00-000 |

Description

VL.CMDU sets the user velocity command. When DRV.OPMODE (→ p. 430) is set to 1 (velocity loop) and DRV.CMDSOURCE (→ p. 375) is set to 0 (TCP/IP channel), then setting this value when the drive is enabled will cause the drive to rotate at the required velocity.

Related Topics

- 1 Velocity Loop

19.30.9 VL.ERR

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Sets the velocity error; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Depends on UNIT.VROTARY (pg 1) or UNIT.VLINEAR (pg 1) Rotary: rpm, rps, deg/s, (custom units)/s, rad/s Linear: counts/s, mm/s, µm/s, (custom units)/s |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| See Also | VL.CMD (→ p. 729), VL.FB (→ p. 732) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3407h/4 | M_01-00-00-000 |

Description

VL.ERR sets the velocity error. It is calculated in the velocity loop as the difference between VL.CMD (→ p. 729) and VL.FB (→ p. 732).

Related Topics

- 1 Velocity Loop

19.30.10 VL.FB

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the velocity feedback; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Depends on UNIT.VROTARY (pg 1) or UNIT.VLINEAR (pg 1) Rotary: rpm, rps, deg/s, (custom units)/s, rad/s Linear: counts/s, mm/s, µm/s, (custom units)/s |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| See Also | VL.CMDU (→ p. 730) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3618h/0 | M_01-00-00-000 |
| | 606Ch/0 | |

Description

VL.FB returns the velocity feedback as it is received in the velocity loop, after passing through Filter 3 and Filter 4.

Related Topics

Velocity Loop

19.30.11 VL.FBFILTER

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Filters VL.FB (→ p. 732) value; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Depends on UNIT.VROTARY (pg 1) or UNIT.VLINEAR (pg 1) Rotary: rpm, rps, deg/s, (custom units)/s, rad/s Linear: counts/s, mm/s, µm/s, (custom units)/s |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| See Also | VL.FB (→ p. 732) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3407h/1 | M_01-00-00-000 |

Description

This parameter returns the same value as VL.FB (→ p. 732), filtered through a 10 Hz filter.

Related Topics

- 1 Velocity Loop

19.30.12 VL.FBSOURCE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets feedback source for the velocity loop; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | PL.FBSOURCE (→ p. 645) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 51AEh/0 | M_01-00-00-000 |

Description

This parameter determines the feedback source to be used by the velocity loop. A value of 0 selects the primary feedback, and 1 selects the secondary feedback.

Related Topics

- 1 Velocity Loop

19.30.13 VL.FBUNFILTERED

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the velocity feedback. |
| Units | Depends on UNIT.VROTARY or UNIT.VLINEAR, UNIT.ACCLINEAR Rotary: rpm, rps, deg/s, (custom units)/s, rad/s Linear: counts/s, mm/s, $\mu\text{m/s}$, (custom units)/s |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| See Also | VL.FB (→ p. 732), VL.FBFILTER (→ p. 733) |
| Start Version | M_01-03-06-000 |

Description

VL.FBUNFILTERED reads the raw velocity feedback before any filters affect the value of this feedback.

Related Topics

Velocity Loop

19.30.14 VL.FF

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Displays the velocity loop overall feedforward value; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Depends on or Rotary: rpm, rps, deg/s, (custom units)/s, rad/s Linear: counts/s, mm/s, μ m/s, (custom units)/s |
| Range | 0 to VL.LIMITP (\rightarrow p. 748) |
| Default Value | 0 |
| Data Type | Float |
| See Also | VL.KBUSFF (\rightarrow p. 739) |
| Start Version | M_01-00-00-000 |

Description

This parameter displays the velocity loop overall feedforward value.

Related Topics

1 Velocity Loop

VL.KVFF (\rightarrow p. 745)

19.30.15 VL.FFDELAY

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the delay time for the velocity feedforward integrator component. |
| Units | Microseconds |
| Range | 0, 100 to 625,000 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-13-01-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 53D6h/0 | M_01-00-00-000 |

Description

VL.FFDELAY sets the delay applied to the velocity feedforward term that is passed into the velocity loop integrator. The delay is applied to the velocity feedforward, and then subtracted from the current velocity feedforward term. This difference is the value fed into the velocity loop integrator. If tuned correctly, this delay will help reduce position following error and velocity following error.

This filter is considered a feedforward. For best results it should be set in conjunction with PL.PDELAY (→ p. 654).

After making an approximation of the Open Loop bandwidth (Hz) in the Performance Servo Tuner, calculate VL.FFDELAY:

$$VL.FFDELAY = 1 / (\text{Open Loop BW (Hz)} * 2\pi) * 1,000,000$$

NOTE

1,000,000 = seconds to microseconds

Related Topics

PL.PDELAY (→ p. 654)

1 Velocity Loop

19.30.16 VL.GENMODE

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Selects mode of velocity generation (Observer, d/dt); active in opmodes 1 (velocity) and 2 (position) only. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 51B0h/0 | M_01-00-00-000 |

Description

This parameter is used to select the velocity generator mode.

| Mode | Description |
|------|--|
| 0 | d/dt mode: The derivative of the mechanical angle of the drive is fed to a first order low pass. |
| 1 | Luenberger Observer mode |

Related Topics

- 1 Velocity Loop

19.30.17 VL.KBUSFF

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the velocity loop acceleration feedforward gain value; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | NA |
| Range | 0.0 to 2.0 |
| Default Value | 0.0 |
| Data Type | Float |
| See Also | VL.BUSFF (→ p. 728) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3407h/3 | M_01-00-00-000 |

Description

This parameter sets the gain for the acceleration feedforward (a scaled second derivative of the position command is added to the velocity command value).

The nominal feedforward value can be multiplied by this gain value.

This will have affect only when using position mode (DRV.OPMODE (→ p. 430) = 2).

Related Topics

- 1 Velocity Loop

19.30.18 VL.KI

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the velocity loop integral gain for the PI controller; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Hz |
| Range | 0 to 1,000 Hz |
| Default Value | 160 Hz |
| Data Type | Float |
| See Also | VL.KP (→ p. 743) |
| Start Version | M_01-00-00-000 |

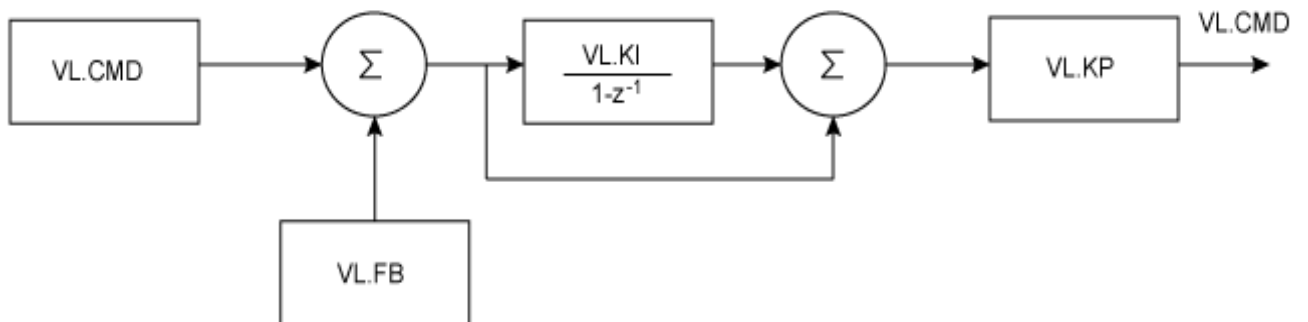
| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 354Dh/0 | M_01-00-00-000 |

Description

VL.KI sets the integral gain of the velocity loop.

A factor of 2π is included in the time calculation, therefore a PI velocity loop with a constant error of 1 rps in which VL.KI is set to 160 and VL.KP (→ p. 743) is set to 1, will take $(1000/160) \cdot 2\pi$ ms to increase the integral gain to 1. Therefore, the total gain is 2 at this time (see velocity loop structure below).

Velocity Loop Structure



Related Topics

- 1 Velocity Loop

19.30.19 VL.KIMODE

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets whether the velocity Integrator is always on, or if the velocity integrator turns off any time the position loop integrator is active. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| Start Version | M_01-14-01-000 |

Description

This parameter sets whether the velocity mode is:

0 – Always on

1 – On ONLY if the position loop integrator is inactive.

PL.KITHRESH (→ p. 652) can be used to set a velocity command threshold to control at what part of a move the position loop integrator is active. When the position loop integrator becomes inactive, the velocity loop integrator will become active.

Common Uses

As a general rule, PL.KI (→ p. 651) should not be used at the same time as VL.KI (→ p. 740), the two integrators will oppose each other and likely cause undesirable results, sometimes a runaway. Using PL.KITHRESH in conjunction with VL.KIMODE the user can specify operating conditions to switch between integrators based on machine operation.

Often times it makes sense to use the velocity integrator during a move, but as the motion comes to an end, to switch back to the position integrator for fine positioning.

Related Topics

VL.KI (→ p. 740) | PL.KITHRESH (→ p. 652)

1 Velocity Loop

19.30.20 VL.KO

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Scales the observer velocity signal; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Internal |
| Range | 0 to 65,535 |
| Default Value | 0 |
| Data Type | Float |
| See Also | VL.MODEL (→ p. 750) |
| Start Version | M_01-00-01-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 51B3h/0 | M_01-00-01-000 |

Description

VL.KO is used to scale the observer model to match the load. When VL.KO is tuned properly, Reads the observer velocity signal; active in opmodes 1 (velocity) and 2 (position) only. (→ p. 750) will match VL.FB (→ p. 732), except when there is an unspecified offset between the two.

Related Topics

- 1 Velocity Loop

19.30.21 VL.KP

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets velocity loop proportional gain for the PI controller; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | A/(rad/sec) |
| Range | 0.001 to 2,147,483.008 |
| Default Value | 1 |
| Data Type | Float |
| See Also | VL.KI (→ p. 740) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3548h/0 | M_01-00-00-000 |

Description

VL.KP sets the proportional gain of the velocity loop.

The idealized velocity loop bandwidth in Hz is:

Rotary motor:

$$\text{Bandwidth (Hz)} = \text{VL.KP} * K_t / (2\pi * J_m)$$

Where:

K_t = motor torque constant, in units of Nm/Arms

J_m = motor inertia, in units of kg*m²

Linear motor:

$$\text{Bandwidth (Hz)} = \text{VL.KP} * K_t / (\text{Motor Pitch (mm)} * J_m)$$

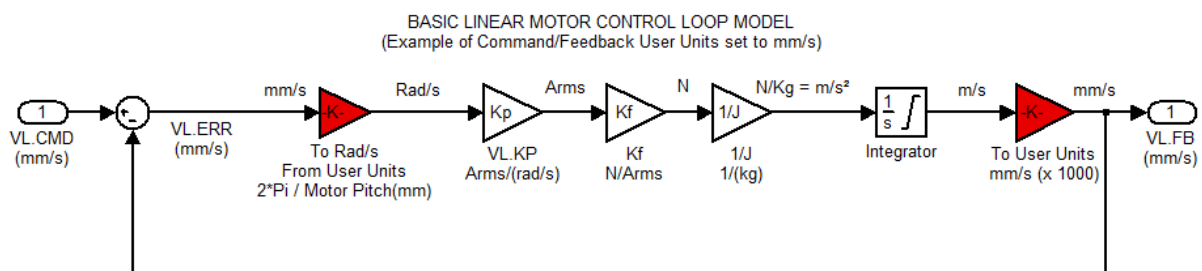
Where:

K_t = motor torque constant, in units of Nm/Arms

J_m = motor inertia, in units of kg

The drive uses the same control loop for both linear and rotary motors. VL.KP units are in Arms/(rad/s). If you want to tune in units of Arms/(mm/s), then you must manually convert the units.

The diagram below shows how linear motors are implemented at the control loop level.



The red blocks are automatically handled at the drive level.

2π radians is the linear equivalent of one full mechanical revolution of a rotary motor - and is equal to the MOTOR.PITCH of a linear motor.

Example

To convert VL.KP = 0.320 Arms/(rad/s) to Arms/(mm/s), where MOTOR.PITCH is 32 mm:

$$VL.KP = 0.320 \text{ Arm /rad/s} * (2\pi \text{ rad} / 32\text{mm MOTOR.PITCH})$$

$$VL.KP = 0.32 * 2\pi / 32 = 0.063 \text{ Arms / (mm/s)}$$

Related Topics

Velocity Controller Environment Block Diagram

19.30.22 VL.KVFF

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the velocity loop velocity feedforward gain value; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | NA |
| Range | 0.0 to 2.0 |
| Default Value | 0.0 |
| Data Type | Float |
| See Also | VL.FF (→ p. 736) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|--------------------|----------------------|
| EtherCAT COE and CANopen | 3407h/2 354Bh/0 | M_01-00-00-000 |

Description

This parameter sets the gain for the velocity feedforward (a scaled derivative of the position command is added to the velocity command value). The nominal feedforward value can be multiplied by this gain value.

This parameter is only used in the position mode (DRV.OPMODE (→ p. 430) = 2).

Related Topics

- 1 Velocity Loop

19.30.23 VL.LIMITN

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the velocity lower limit; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Depends on UNIT.VROTARY (pg 1) or UNIT.VLINEAR (pg 1) Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: counts/s, mm/s, $\mu\text{m/s}$, custom units/s |
| Range | Range may be limited by ECCN 3A225 Limitations for Induction Motors (→ p. 1) Rotary: -15,000.000 to 0.000 rpm -250.000 to 0.000 rps -90,000.000 to 0.000 deg/s -1,250.000 to 0.000 custom units/s -1570.796 to 0.000 rad/s Linear: -1,073,741,824,000.000 to 0.000 counts/s -250.000*MOTOR.PITCH (→ p. 617) to 0.000 mm/s -250,000.000*MOTOR.PITCH (→ p. 617) to 0.000 $\mu\text{m/sec}$ -1,250.000 to 0.000 custom units/s |
| Default Value | Rotary: -3,000.000 rpm -50.000 rps -18,000.002 deg/s -250.000 (custom units)/s -314.159 rad/s Linear: -0.050 counts/s -50*MOTOR.PITCH (→ p. 617) mm/s -50,000.004*MOTOR.PITCH $\mu\text{m/sec}$ -250.000 custom units/s |
| Data Type | Float |
| See Also | VL.LIMITP (→ p. 748), VL.CMD (→ p. 729) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3623h/0 | M_01-00-00-000 |

Description

VL.LIMITN sets the velocity command negative limit.

If the input to the velocity loop is lower than VL.LIMITN, then the actual velocity command VL.CMD (→ p. 729) is limited by the value of VL.LIMITN.

NOTE

ECCN 3A255 imposes special limitations on operating speed for induction motors. See ECCN 3A225 Limitations for Induction Motors (→ p. 1).

Related Topics

- 1 Velocity Loop

19.30.24 VL.LIMITP

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the velocity high limit; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Depends on UNIT.VROTARY (pg 1) or UNIT.VLINEAR (pg 1) Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: counts/s, mm/s, μ m/s, custom units/s |
| Range | Range may be limited by ECCN 3A225 Limitations for Induction Motors (\rightarrow p. 1) Rotary: 0.000 to 15,000.000 rpm 0.000 to 250.000 rps 0.000 to 90,000.000 deg/s 0.000 to 1,250.000 custom units/s 0.000 to 1570.796 rad/s Linear: 0.000 to 1,073,741,824,000.000 counts/s 0.000 to 250.000*MOTOR.PITCH (\rightarrow p. 617) mm/sec 0.000 to 250,000.000*MOTOR.PITCH (\rightarrow p. 617) μ m/s 0.000 to 1,250.000 custom units/s |
| Default Value | Rotary: 3,000.000 rpm 50.000 rps 18,000.002 deg/s 250.000 (custom units)/s 314.159 rad/s Linear: 0.050 counts/s 50.000*MOTOR.PITCH (\rightarrow p. 617) mm/sec 50,000.004*MOTOR.PITCH μ m/sec 250.000 custom units/s |
| Data Type | Float |
| See Also | VL.LIMITN (\rightarrow p. 746), VL.CMD (\rightarrow p. 729) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3622h/0 | M_01-00-00-000 |

Description

VL.LIMITP sets the velocity command positive limit.

If the input to the velocity loop is higher than VL.LIMITP, then the actual velocity command VL.CMD (\rightarrow p. 729) is limited by the value of VL.LIMITP.

NOTE

ECCN 3A255 imposes special limitations on operating speed for induction motors. See ECCN 3A225 Limitations for Induction Motors (\rightarrow p. 1).

Related Topics

- 1 Velocity Loop

19.30.25 VL.LMJR

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the ratio of the estimated load moment of inertia relative to the motor moment of inertia; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | NA |
| Range | 0 to 100.0 |
| Default Value | 0 |
| Data Type | Float |
| See Also | IL.FF (→ p. 557) |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 51B8h/0 | M_01-00-00-000 |

Description

This parameter is used in the internal calculation of the current loop acceleration feed forward gain value.

Related Topics

- 1 Velocity Loop

19.30.26 VL.MODEL

| General Information | |
|---------------------|---|
| Type | R/O Parameter |
| Description | Reads the observer velocity signal; active in opmodes 1 (velocity) and 2 (position) only. |
| Units | Depends on UNIT.VROTARY or UNIT.VLINEAR, UNIT.ACCLINEAR Rotary: rpm, rps, deg/s, (custom units)/s, rad/s Linear: counts/s, mm/s, $\mu\text{m/s}$, (custom units)/s |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| See Also | VL.FB (→ p. 732), VL.KO (→ p. 742) |
| Start Version | M_01-00-01-000 |

Description

VL.MODEL is the observer velocity output. When VL.KO (→ p. 742) is tuned properly, VL.MODEL will match VL.FB (→ p. 732), except when there is an unspecified offset between the two.

Related Topics

- 1 Velocity Loop

19.30.27 VL.OBSBW

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the bandwidth of the observer in Hz. |
| Units | Hz |
| Range | 10 to 4,000 Hz |
| Default Value | 30 Hz |
| Data Type | Float |
| See Also | N/A |
| Start Version | M_01-03-00-004 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 51BAh/0 | M_01-03-00-004 |

Description

This parameter sets the bandwidth of the observer in Hz. The observer passes the velocity feedback through a PID control loop that behaves like a low-pass filter with a bandwidth of VL.OBSBW.

Related Topics

- 1 Velocity Loop

19.30.28 VL.OBSMODE

| General Information | |
|---------------------|-----------------------------------|
| Type | NV Parameter |
| Description | Sets the observer operating mode. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | Integer |
| See Also | N/A |
| Start Version | M_01-03-00-004 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 51BBh/0 | M_01-03-00-004 |

Description

This parameter sets the observer operating mode. When VL.OBSMODE = 0, the observer is not part of the control loop; that is, VL.FB is used as the velocity feedback signal to the velocity loop. When VL.OBSMODE = 1, the observer is part of the control loop; VL.MODEL is used as the velocity feedback signal.

Related Topics

- 1 Velocity Loop

19.30.29 VL.THRESH

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the user over speed fault level. |
| Units | Depends on UNIT.VROTARY (pg 1) or UNIT.VLINEAR (pg 1) Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: counts/s, mm/s, μ m/s, custom units/s |
| Range | Rotary: 0.000 to 15,000.000 rpm 0.000 to 250.000 rps 0.000 to 90,000.000 deg/s 0.000 to 1,250.000 custom units/s 0.000 to 1,570.796 rad/s Linear: 0.000 to 1,073,741,824,000.000 counts/s 0.000 to 250.000*MOTOR.PITCH (\rightarrow p. 617) mm/s 0.000 to 250,000.000*MOTOR.PITCHMOTOR.PITCH (\rightarrow p. 617) μ m/s 0.000 to 1,250.000 custom units/s |
| Default Value | Rotary: 3,600 rpm 60 rps 21,600.000 deg/s 300.000 custom units/s 376.991 rad/s Linear: 0.060 counts/s 60.000*MOTOR.PITCH (\rightarrow p. 617) mm/s 60,000.04*MOTOR.PITCHMOTOR.PITCH (\rightarrow p. 617) μ m/s 300.000 custom units/s |
| Data Type | Float |
| Start Version | M_01-00-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3627h/0 | M_01-00-00-000 |

Description

VL.THRESH sets the user overspeed fault threshold. This threshold becomes the overall overspeed fault threshold (VL.VFTHRESH) if it is less than the motor speed threshold ($1.2 \times \text{MOTOR.VMAX}$), and for MOTOR.TYPE = 6 if it is less than the motor back EMF overspeed threshold ($(1.02 \times 1000 \times \text{VBUS.OVFTHRESH}) / (\text{MOTOR.KE} \times \sqrt{2})$).

The value is considered an absolute value, hence it applies for both negative and positive velocities.

Example

VL.THRESH is set to 600 rpm. A velocity (VL.FB (\rightarrow p. 732)) of 700 rpm will generate an over speed fault.

Related Topics

1 Velocity Loop | VL.CMD (→ p. 729) | VL.CMDU (→ p. 730) | VL.VFTHRESH (→ p. 755)

19.30.30 VL.VFTHRESH

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads the overall overspeed fault threshold. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | Float |
| Start Version | M_01-10-03-000 |

Description

This parameter reports the overall overspeed fault threshold. Whichever of the following thresholds has the smallest value is used as the overall overspeed fault threshold:

- User overspeed threshold (VL.THRESH).
- Motor mechanical overspeed threshold ($1.2 \times \text{MOTOR.VMAX}$)
- If $\text{MOTOR.TYPE} = 6$, the motor back EMF overspeed threshold ($((1.02 \times 1000 \times \text{VBUS.OVFTHRESH}) / (\text{MOTOR.KE} \times \sqrt{2}))$).

Related Topics

1 Velocity Loop | VL.THRESH (→ p. 753) | MOTOR.VMAX (→ p. 631)

19.31 WS Parameters

This section describes the WS parameters.

| | |
|--------------------------------------|------------|
| 19.31.1 WS.ARM | 757 |
| 19.31.2 WS.CHECKMODE | 758 |
| 19.31.3 WS.CHECKT | 759 |
| 19.31.4 WS.CHECKV | 760 |
| 19.31.5 WS.DISARM | 761 |
| 19.31.6 WS.DISTMAX | 762 |
| 19.31.7 WS.DISTMIN | 763 |
| 19.31.8 WS.FREQ | 764 |
| 19.31.9 WS.IMAX | 765 |
| 19.31.10 WS.MODE | 766 |
| 19.31.11 WS.NUMLOOPS | 767 |
| 19.31.12 WS.STATE | 768 |
| 19.31.13 WS.T | 769 |
| 19.31.14 WS.TDELAY1 | 770 |
| 19.31.15 WS.TDELAY2 | 771 |
| 19.31.16 WS.TDELAY3 | 772 |
| 19.31.17 WS.TDELAY4 | 773 |
| 19.31.18 WS.TIRAMP | 774 |
| 19.31.19 WS.TSTANDSTILL | 775 |
| 19.31.20 WS.VTHRESH | 776 |

19.31.1 WS.ARM

| General Information | |
|---------------------|--|
| Type | Command |
| Description | Sets wake and shake to start at the next drive enable. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| See Also | N/A |
| Start Version | M_01-01-00-101, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3494h/6 | M_01-00-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

This command sets wake and shake to start at the next drive enable. Feedback type is not relevant for this command. If WS.STATE is 0 and the drive is disabled, then WS.STATE will change to 1 after issuing WS.ARM. With this command, wake and shake can be repeated if desired.

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.2 WS.CHECKMODE

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Select the type of commutation check to execute after Wake and Shake finds a new commutation angle. |
| Units | N/A |
| Range | 0 to 2 |
| Default Value | 1 |
| Data Type | Integer |
| Start Version | M_01-07-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3494h/8 | M_01-00-00-000 |

Description

AKD PDMM supports several options to confirm a valid commutation after Wake and Shake has determined a new commutation angle. In some mechanical situations, such as if motion is inhibited, it is possible for Wake and Shake to select an incorrect angle. Commutation checks are designed to prevent unexpected motion if this occurs.

0 = No Commutation Check

If No Commutation Check is selected, neither passive nor active commutations checks will be executed.

1 = Active Commutation Check (default)

In the default Active Commutation Check mode, AKD PDMM will make a short torque move after an angle has been selected. If the motor fails to move in the expected direction, a fault will be generated.

2 = Passive Commutation Check

In the Passive Commutation Check mode, AKD PDMM will monitor torque commands and acceleration values for 10 revolutions after Wake and Shake has completed. During this time, a fault will be generated if unexpected motion is detected, such as if commanded torque and acceleration are in opposition directions. Note that this fault may also be reported if the motor experiences a large torque disturbance lasting longer than WS.CHECKT.

Related Topics

Wake and Shake Overview (→ p. 56) | WS.MODE (→ p. 766)

19.31.3 WS.CHECKT

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the amount of time a commutation error must be present before an error is thrown. |
| Units | ms |
| Range | 0 to 10,000 |
| Default Value | N/A |
| Data Type | U16 |
| Start Version | M_01-06-03-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 5352h/0 | M_01-06-03-000 |

Description

This parameter sets the amount of time a commutation error must be present before an error is thrown. Commutation monitoring is disabled when this parameter is set to 0.

Related Topics

Wake and Shake Overview (→ p. 56) | WS.CHECKV (→ p. 760)

19.31.4 WS.CHECKV

| General Information | | | |
|---------------------|---|--------------------|--------------------------|
| Type | R/W Parameter | | |
| Description | This parameter sets the velocity threshold which has to be exceeded to activate commutation monitoring. | | |
| Units | Depends on UNIT.VROTARY or UNIT.VLINEAR. | | |
| Range | Unit | Rotary | Linear |
| | 0 | 0 to 15,000 rpm | 0 to 6,7108,864 counts/s |
| | 1 | 0 to 250 rps | 0 to 8,000 mm/s |
| | 2 | 0 to 90,000 deg/s | 0 to 589934 um/s |
| | 3 | 0 to 1250 PIN/POUT | 0 to 1250 PIN/POUT |
| Default Value | N/A | | |
| Data Type | S32 | | |
| Start Version | M_01-06-03-000 | | |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3494h/7 | M_01-00-00-000 |

Description

This parameter sets the velocity threshold which has to be exceeded to activate commutation monitoring.

Related Topics

Wake and Shake Overview (→ p. 56) | WS.CHECKT (→ p. 759)

19.31.5 WS.DISARM

| General Information | |
|---------------------|---|
| Type | Command |
| Description | Cancels ARM requests and resets wake and shake to the IDLE state. |
| Units | N/A |
| Range | N/A |
| Default Value | N/A |
| Data Type | N/A |
| See Also | N/A |
| Start Version | M_01-04-00-000 |

*For Profinet, write a value of 0 into the PNU to execute a command.

Description

This command disables wake and shake immediately. Feedback type is not relevant for this command. If WS.ARM (→ p. 757) has been issued, the request to execute the wake and shake algorithm at the next enable is cancelled. WS.STATE (→ p. 768) is set to IDLE.

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.6 WS.DISTMAX

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets maximum movement allowed for wake and shake. |
| Units | deg (position units) |
| Range | 0 to 90 deg |
| Default Value | 15 deg |
| Data Type | S64 |
| See Also | N/A |
| Start Version | M_01-01-00-101, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3494h/2 | M_01-00-00-000 |

Description

This parameter sets the maximum movement that is allowed for finding commutation. If this value is too small, FF475 (→ p. 187), "Wake and Shake. Too much movement", may occur before wake and shake is finished. The bigger this value, the more movement is allowed for wake and shake. This value is application dependent.

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.7 WS.DISTMIN

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the minimum movement required for wake and shake. |
| Units | Actual position units |
| Range | 0 to 90 deg |
| Default Value | 1 deg |
| Data Type | S64 |
| See Also | N/A |
| Start Version | M_01-01-00-101, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 36D1h/0 | M_01-00-00-000 |

Description

This parameter sets the minimum movement that is required for commutation finding. If this value is too small, the the commutation finding might fail if too little current is used. The larger this value, the more movement is needed in order to avoid F473: "Wake and Shake: Too little movement".

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.8 WS.FREQ

| General Information | |
|---------------------|--|
| Type | R/W |
| Description | Sets the sine frequency of excitation for WS.MODE 2. |
| Units | Hz |
| Range | 0.01 - 8000 |
| Default Value | 10 |
| Data Type | Float |
| Start Version | M_01-05-10-000 |

Description

This keyword sets the excitation frequency for Wake and Shake mode 2.

The default setting of 10Hz will work for almost all systems.

In cases where the load attached to the motor is much higher than the motor inertia (200:1) it may be necessary to lower the value of WS.FREQ.

Effects of lowering WS.FREQ:

- Wake and Shake Mode 2 will take longer to complete.
- A higher magnitude of motion will be seen on the motor because the motor has current applied in the same direction for a longer period of time.

Related Topics

WS.MODE (→ p. 766)

19.31.9 WS.IMAX

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets maximum current used for wake and shake. |
| Units | Arms |
| Range | 0 to (lower value of MOTOR.IPEAK and DRV.IPEAK) Arms |
| Default Value | (half of maximum) Arms |
| Data Type | U16 |
| See Also | MOTOR.IPEAK (→ p. 607), DRV.IPEAK (→ p. 417) |
| Start Version | M_01-01-00-101, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3494h/1 | M_01-00-00-000 |

Description

This parameter defines the maximum current used for wake and shake. If the selected current is too low, the minimum required movement may not occur. If the selected current is too high, the movement may be too fast (overspeed) or too large (over maximum movement).

The maximum of this parameter is the lower value of MOTOR.IPEAK and DRV.IPEAK. The default value of this parameter is the half of its maximum. This value depends on the specific application.

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.10 WS.MODE

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the method used for wake and shake. |
| Units | N/A |
| Range | 0 to 1 |
| Default Value | 0 |
| Data Type | U8 |
| Start Version | M_01-01-00-101, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3874/0 | M_01-00-00-000 |

Description

This parameter sets the method used for finding commutation.

0 = Standard wake and shake

Two iterations are used to find the correct angle in this mode. Coarse (current mode) and fine (velocity mode) iterations are done in a loop (WS.NUMLOOPS times). The average angle of all loops is calculated and used.

1 = Commutation alignment by fixed commutation vector (Zero Method)

The motor poles are set to 0, current mode is activated, and WS.IMAX is applied. The angle in which the motor settles is used for commutation. Other settings are restored (such as motor poles and operation mode).

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.11 WS.NUMLOOPS

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the number of repetitions for wake and shake. |
| Units | counts |
| Range | 0 to 20 counts |
| Default Value | 5 counts |
| Data Type | U8 |
| See Also | N/A |
| Start Version | M_01-01-00-101, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 36E2h/0 | M_01-00-00-000 |

Description

This parameter sets the maximum number of wake and shake repetitions. MOTOR.PHASE is calculated as mean value of all wake and shake repetitions.

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.12 WS.STATE

| General Information | |
|---------------------|--|
| Type | R/O Parameter |
| Description | Reads wake and shake status. |
| Units | N/A |
| Range | N/A |
| Default Value | Only valid before the first enable occurs. 11 - for feedback types that do not require wake and shake 1 - for feedback types that require wake and shake |
| Data Type | U8 |
| See Also | N/A |
| Start Version | M_01-01-00-101, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3494h/5 | M_01-00-00-000 |

Description

WS switches different current vectors and records position feedback in order to establish commutation alignment.

WS.STATE 0 = wake and shake successful (DONE).

WS.STATE 1 = wake and shake configured and will be done at next enable (ARMED).

WS.STATE 2 = wake and shake running. (ACTIVE)

WS.STATE 10 = error occurred during wake and shake (ERROR).

WS.STATE 11 = wake and shake not required (IDLE).

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.13 WS.T

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets wake and shake current-vector appliance time. |
| Units | ms |
| Range | 1 to 200 ms |
| Default Value | 2 ms |
| Data Type | U8 |
| See Also | WS.IMAX (→ p. 765), WS.DISTMAX |
| Start Version | M_01-01-00-101, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 36D0h/0 | M_01-00-00-000 |

Description

This parameter defines the duration for each different current-vector while the coarse angle calculation. The move distance is proportional to the WS.T and WS.IMAX value.

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.14 WS.TDELAY1

| General Information | |
|---------------------|----------------------------------|
| Type | NV Parameter |
| Description | Delay for wake and shake timing. |
| Units | ms |
| Range | 0 to 200 ms |
| Default Value | 5 ms |
| Data Type | U8 |
| See Also | N/A |
| Start Version | M_01-01-00-101, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3683h/0 | M_01-00-00-000 |

Description

WS.TDELAY1 defines the delay time of the wake and shake function. This time is a delay time between the switching of different current vectors during the wake and shake procedure. This time should be increased in the case of movement interferences between single current vectors.

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.15 WS.TDELAY2

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Sets the delay for wake and shake timing. |
| Units | ms |
| Range | 0 to 200 ms |
| Default Value | 50 ms |
| Data Type | U8 |
| See Also | N/A |
| Start Version | M_01-01-00-101, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3685h/0 | M_01-00-00-000 |

Description

WS.TDELAY2 defines the delay between switching from coarse angle calculation to fine angle calculation during the wake and shake procedure. This time should be increased in the case of interferences between the coarse calculation done in current mode and the fine calculation done in velocity mode. Choosing too large a value increases the wake and shake duration.

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.16 WS.TDELAY3

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the delay for wake and shake between loops in mode 0. |
| Units | ms |
| Range | 0 to 2,000 ms |
| Default Value | 100 ms |
| Data Type | U16 |
| See Also | N/A |
| Start Version | M_01-01-00-102, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3494h/3 | M_01-00-00-000 |

Description

WS.TDELAY3 defines the delay between complete loops in mode 0 only. Decreasing this value makes the wake and shake procedure faster, but may lead to problems if the motor moves too long. Increasing this value will make wake and shake significantly longer.

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.17 WS.TDELAY4

| General Information | |
|---------------------|--|
| Type | NV Parameter |
| Description | Sets the delay time after first current-pulse injection. |
| Units | ms |
| Range | 0 to 10,000 ms |
| Default Value | 2000 ms |
| Data Type | U16 |
| Start Version | M_01-01-00-102, M_01-02-00-000 |

Description

WS.TDELAY4 defines the delay time after first current-pulse injection.

Related Topics

Wake and Shake Overview (→ p. 56)

19.31.18 WS.TIRAMP

| General Information | |
|---------------------|--|
| Type | R/W Parameter |
| Description | Sets the ramp time for the ramp up current in Wake & Shake mode 1. |
| Units | ms |
| Range | 512 to 10,000 |
| Default Value | 512 |
| Data Type | U16 |
| Start Version | M_01-06-07-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 535Dh/0 | M_01-06-07-000 |

Description

In Wake & Shake mode 1, a maximum current of WS.IMAX is applied to the motor. WS.TIRAMP defines a time to reach this maximum current. Longer times are useful for big inertia motors or heavy loads.

Related Topics

Wake and Shake Overview (→ p. 56) | WS.MODE (→ p. 766) | WS.TSTANDSTILL (→ p. 775)

19.31.19 WS.TSTANDSTILL

| General Information | |
|---------------------|---|
| Type | R/W Parameter |
| Description | Sets the calming time of the motor for Wake & Shake mode 1. |
| Units | ms |
| Range | 100 to 20,000 |
| Default Value | 1000 |
| Data Type | U16 |
| Start Version | M_01-06-07-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 535Ch/0 | M_01-06-07-000 |

Description

In Wake & Shake mode 1, a maximum current of WS.IMAX is applied to the motor. WS.TSTANDSTILL defines a wait time, during which this maximum current is held, before detecting the commutation angle of the motor. Longer times are useful for big inertia motors or heavy loads.

Related Topics

Wake and Shake Overview (→ p. 56) | WS.MODE (→ p. 766) | WS.TIRAMP (→ p. 774)

19.31.20 WS.VTHRESH

| General Information | |
|---------------------|---|
| Type | NV Parameter |
| Description | Defines the maximum allowed velocity for Wake & Shake. |
| Units | Rotary: rpm, rps, deg/s, rad/s Linear: counts/s, mm/s, μ m/s |
| Range | Rotary: 0.000 to 15,000.000 rpm 0.000 to 250.000 rps 0.000 to 90,000.000 degree/s 0.000 to 1,570.796 rad/s Linear: 0.000 to 1,073,741,824,000.000 counts/s 0.000 to 8,000.000 mm/s 0.000 to 8,000,000.000 μ m/s |
| Default Value | 100 rpm |
| Data Type | U16 |
| See Also | N/A |
| Start Version | M_01-01-00-101, M_01-02-00-000 |

| Fieldbus | Index/Subindex | Object Start Version |
|--------------------------|----------------|----------------------|
| EtherCAT COE and CANopen | 3494h/4 | M_01-00-00-000 |

Description

This parameter defines the maximum allowed velocity that occurs while commutation finding is active. This supervision runs in real time, but only while wake and shake is active (WS.STATE 2 or greater, for Mode 0). If at any time while wake and shake is running a velocity higher than this value is detected, fault F478n478 (→ p. 188) is generated. Setting WS.VTHRESH to zero disables this feature. For Mode 1, WS.VTHRESH is only used after the initial phase-finding.

Related Topics

Wake and Shake Overview (→ p. 56)

This page intentionally left blank.

19.32 EtherCAT Profile

19.32.1 Slave Register

The table below gives the addresses of individual registers in the FPGA memory. The data is provided in little-endian format, with the 'least significant byte' occupying the lowest address. A detailed description of all registers and FPGA memory locations is available in the "EtherCAT Slave Controller" description of the EtherCAT user organization (www.EtherCAT.org).

| Address | Length (Byte) | Description | ZA ECAT* | ZA Drive* |
|---------|---------------|---|----------|-----------|
| 0x0120 | 2 | AL Control | R/W | R/O |
| 0x0130 | 2 | AL Status | R/O | R/W |
| 0x0134 | 2 | AL Status Code | R/O | R/W |
| 0x0204 | 2 | Interrupt Enable Register | R/O | R/W |
| 0x0220 | 2 | AL Event (IRQ Event) | R/W | R/O |
| 0x0800 | 8 | Sync Manager 0 (Mail Out Control Register) | R/W | R/O |
| 0x0808 | 8 | Sync Manager 1 (Mail In Control Register) | R/W | R/O |
| 0x0810 | 8 | Sync Manager 2 (Process data Output Control Register) | R/W | R/O |
| 0x0818 | 8 | Sync Manager 3 (Process data Input Control Register) | R/W | R/O |
| 0x0820 | 8 | Sync Manager 4 | R/W | R/O |
| 0x0828 | 8 | Sync Manager 5 | R/W | R/O |
| 0x0830 | 8 | Sync Manager 6 | R/W | R/O |
| 0x0838 | 8 | Sync Manager 7 | R/W | R/O |
| 0x0840 | 8 | Sync Manager 8 | R/W | R/O |
| 0x1100 | Max. 64 | ProOut Buffer (Process data Output, set-points ECAT) | R/W | R/O |
| 0x1140 | Max. 64 | ProIn (Process data Input, act. values ECAT) | R/O | R/W |
| 0x1800 | 512 | Mail Out Buffer (Object Channel Buffer ECAT, byte-length is specified in the device description file) | R/W | R/O |
| 0x1C00 | 512 | Mail In Buffer (Object Channel Buffer Drive, byte-length is specified in the device description file) | R/O | R/W |

* ZA ECAT = Access mode EtherCAT

* ZA Drive = Access mode drive

19.32.2 AL Event (Interrupt Event) and Interrupt Enable

Communication between the drive and the EtherCAT FPGA can be interrupt-driven. The interrupt enable register and the AL event register are responsible for the EtherCAT interface interrupt functionality.

There are two events which lead also to a HW interrupt within the drive, the EEPROM emulation event and the SyncManager 2 event. The actual values of the drive (SyncManager 3 data) are written without any AL event request during each HW IRQ, e.g. triggered by a SyncManager 2 event. The Mailbox exchange between the master and the AKD PDMM is completely handled by polling the AL event register within the background task of the drive.

The drive activates individual EtherCAT interface events when the corresponding bit of the interrupt enable register is set to 1. When it is set to 0, the hardware interrupts for the specific events are deactivated.

19.32.2.1 Interrupt Enable Register (Address 0x0204:0x0205)

| Parameter | Address | Bit | ZA Drive | ZA ECAT | Description |
|--|---------|--------|----------|---------|---|
| AL Control Event | 0x204 | 0 | R/W | R/O | Activation of AL control event for phase run-up |
| - | 0x204 | 1 | R/W | R/O | Reserved |
| Sync0 DC Distributed Clock | 0x204 | 2 | R/W | R/O | Activation of distributed clock (DC) sync 0 interrupts for entire communication |
| Sync1 DC Distributed Clock | 0x204 | 3 | R/W | R/O | Activation of distributed clock (DC) sync 1 interrupts for entire communication |
| SyncManager activation register change | 0x204 | 4 | R/W | R/O | Activation of 'SyncManager activation register change' IRQ. |
| EEPROM emulation event | 0x204 | 5 | R/W | R/O | Activation of the EEPROM emulation interrupts. |
| - | 0x204 | 3 to 7 | R/W | R/O | Reserved |
| Sync Manager 0 Event (Mail Out Event) | 0x205 | 0 | R/W | R/O | Activation of output event mailbox (SDO, Sync Manager 0) for object channel. |
| Sync Manager 1 Event (Mail In Event) | 0x205 | 1 | R/W | R/O | Activation of input event mailbox (SDO, Sync Manager 1) for object channel. |
| Sync Manager 2 Event (Pro Out Event) | 0x205 | 2 | R/W | R/O | Activation of output event process data (PDO, card's cyclical setpoints) |
| Sync Manager 3 Event (Pro In Event) | 0x205 | 3 | R/W | R/O | Activation of input event process data (PDO, drive's cyclical actual values) |
| - | 0x205 | 4 to 7 | R/W | R/O | Reserved |

19.32.2.2 AL Event Request (Address 0x0220:0x0221)

When the relevant bit of the AL event request register is set to 1, the EtherCAT interface tells the drive which event it should process by the AKD PDMM.

| Parameter | Address | Bit | ZA Drive | ZA ECAT | Description |
|--|---------|--------|----------|---------|---|
| AL Control Event | 0x220 | 0 | R/O | R/W | Processing of AL control event for phase run-up |
| Sync0 Distributed Clock (DC) Event | 0x220 | 2 | R/O | R/W | Processing of a distributed clock (DC) event |
| Sync1 Distributed Clock (DC) Event | 0x220 | 3 | R/O | R/W | Processing of a distributed clock (DC) event |
| SyncManager activation register change | 0x220 | 4 | R/O | R/W | The content of the SyncManager activation register has been changed. |
| EEPROM emulation event | 0x220 | 5 | R/O | R/W | Processing of an EEPROM emulation event in order to identify the AKD PDMM within the network. |
| - | 0x220 | 6 to 7 | R/O | R/W | Reserved |
| Sync Manager 0 Event | 0x221 | 0 | R/O | R/W | Mailbox request (SDO, Sync Manager 0) for object channel. |
| Sync Manager 1 Event | 0x221 | 1 | R/O | R/W | Mailbox response (SDO, Sync Manager 1) for object channel. |
| Sync Manager 2 Event | 0x201 | 2 | R/O | R/W | Process data output (PDO, card's cyclical setpoints) |
| Sync Manager 3 Event | 0x201 | 3 | R/O | R/W | Process data input (PDO, drive's cyclical actual values) |
| Sync Manager 4 – | | | | | |
| Sync Manager 7 Event | 0x221 | 4 to 7 | R/O | R/W | Reserved |
| Sync Manager 8 – | | | | | |
| Sync Manager 15 Event | 0x222 | 0 to 7 | R/O | R/W | Reserved |

19.32.3 Phase Run-Up

The AL control, AL status and AL status code registers are responsible for communication phase run-up (also referred to as EtherCAT status change), for current status display and for any fault messages. The drive responds to every EtherCAT interface transition request made by the AL control register via the AL Status and AL Status Code registers. Any fault messages are displayed in the AL status code register.

A status change within the AL control register is polled within the AKD PDMM, which means that an AL control event does not lead to a HW interrupt within the drive.

19.32.3.1 AL Control (Address 0x0120:0x0121)

| Parameter | Address | Bit | ZA Drive | ZA ECAT | Description |
|--------------------------------|---------|---------|----------|---------|---|
| Status | 0x120 | 3 to 0 | R/O | W/O | 0x01: Init Request |
| 0x02: PreOperational Request | | | | | |
| 0x03: Bootstrap Mode Request | | | | | |
| 0x04: Safe Operational Request | | | | | |
| 0x08: Operational Request | | | | | |
| Acknowledgement | 0x120 | 4 | R/O | W/O | 0x00: No fault acknowledgement 0x01: Fault acknowledgement (positive edge) |
| Reserved | 0x120 | 7 to 5 | R/O | W/O | - |
| Applic. specific | 0x120 | 15 to 8 | R/O | W/O | - |

19.32.3.2 AL Status (Address 0x0130:0x0131)

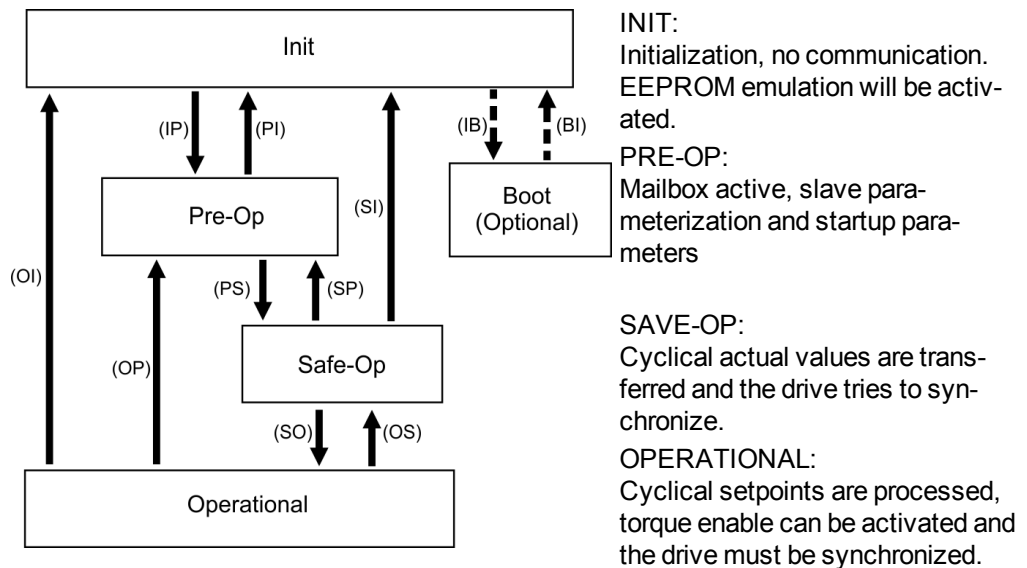
| Parameter | Address | Bit | ZA Drive | ZA ECAT | Description |
|------------------------|---------|---------|----------|---------|---|
| Status | 0x130 | 3 to 0 | W/O | R/O | 0x01: Init |
| 0x02: PreOperational | | | | | |
| 0x03: Bootstrap Mode | | | | | |
| 0x04: Safe Operational | | | | | |
| 0x08: Operational | | | | | |
| Status change | 0x130 | 4 | W/O | R/O | 0x00: Acknowledgement 0x01: Error, e.g. forbidden transition |
| Reserved | 0x130 | 7 to 5 | W/O | R/O | - |
| Applic. specific | 0x130 | 15 to 8 | W/O | R/O | - |

19.32.3.3 AL Status Code (Address 0x0134:0x0135)

| Parameter | Address | Bit | ZA Drive | ZA ECAT | Description |
|-----------|------------------------------------|--|----------|--------------------|-----------------|
| Status | 0x134 | 7 to 0 | W/O | R/O | See table below |
| Status | 0x135 | 7 to 0 | W/O | R/O | See table below |
| Code | Description | Current Status (Status change) | | Resulting Status | |
| 0x0000 | No error | All | | Current Status | |
| 0x0011 | Invalid requested state change | I -> S, I -> O, P -> O, O -> B, S -> B, P -> B | | Current Status + E | |
| 0x0017 | Invalid sync manager configuration | I -> P, P -> S | | Current Status + E | |

No other codes are supported.

19.32.3.4 EtherCAT communication phases

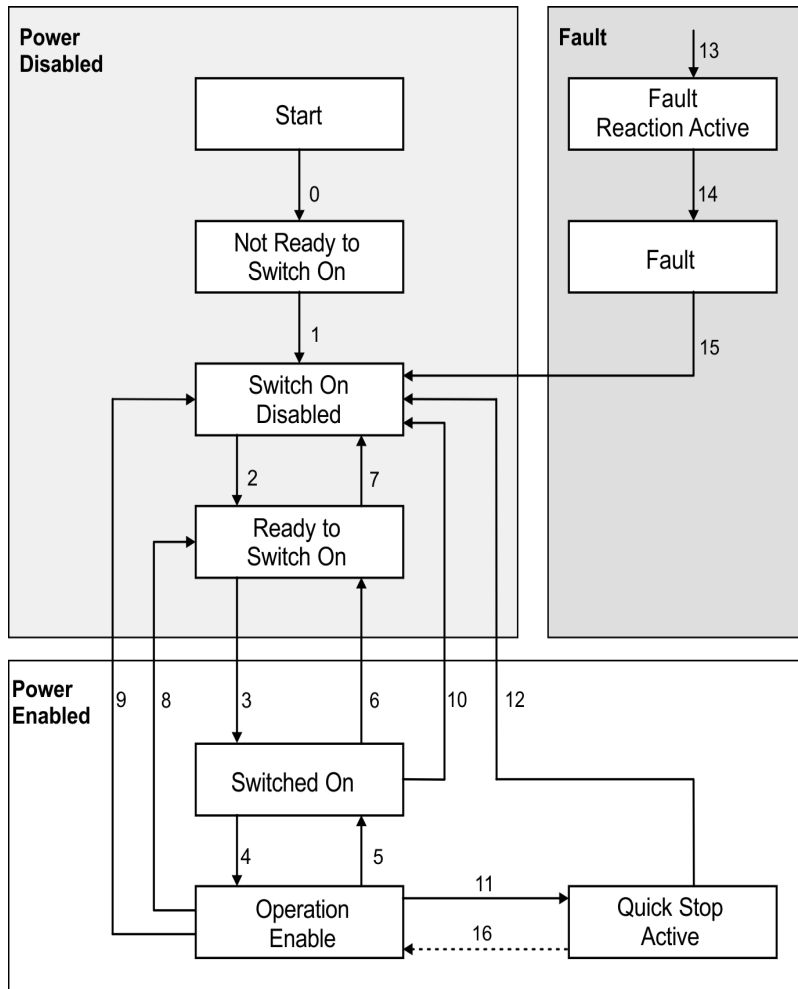


Individual communication transitions

| Transition | AL Control (Bit 3 to 0) | Description |
|------------|-------------------------|--|
| (IB) | 0x03 | - |
| (BI) | - | - |
| (IP) | 0x02 | AKD PDMM reads the SyncManager 0 & 1 configuration and verifies the value of the start-address and the length. The AKD PDMM prepares itself for handling SyncManager 0 events. |
| (PI) | 0x01 | - |
| (PS) | 0x04 | AKD PDMM reads the SyncManager 2 & 3 configuration and verifies the value of the start-address and the length. |
| (SP) | 0x02 | - |
| (SI) | 0x01 | - |

| Transition | AL Control (Bit 3 to 0) | Description |
|------------|-------------------------|---|
| (SO) | 0x08 | The SsyncManager 2 hardware interrupt will be enabled by the drive. |
| (OS) | 0x04 | Deactivation of SsyncManager 2 hardware interrupt. |
| (OP) | 0x02 | Deactivation of SsyncManager 2 hardware interrupt.. |
| (OI) | 0x01 | Deactivation of SsyncManager 2 hardware interrupt. |

19.32.4 CANopen over EtherCAT (CoE) State Machine



The state machine for the control and status words corresponds to the CANopen state machine in accordance with DS402.

CANopen control and status words are captured in every instance of fixed PDO mapping (see chapter "Fixed PDO Mappings" (→ p. 943)).

19.32.4.1 Status Description

| Status | Description |
|------------------------|--|
| Not Ready to Switch On | The drive is not ready to switch on; the controller has not indicated readiness for service. The drive is still in the boot phase or in fault status. |
| Switch On Disable | In 'Switch On Disable' status, the amplifier cannot be enabled via the EtherCAT interface, because (for example) there is no connection to a power source. |
| Ready to Switch On | In 'Ready to Switch On' status, the drive can be enabled via the control word. |
| Switched On | In 'Switched On' status, the amplifier is enabled, but the setpoints of the EtherCAT-interface are not yet transferred. The amplifier is idle, and a positive edge in bit 3 of the control word activates setpoint transfer (transition to 'Operation Enable' status). |
| Operation Enable | In this status, the drive is enabled and setpoints are transferred from the EtherCAT interface. |
| Quick Stop Active | The drive follows a quick stop ramp. |
| Fault Reaction Active | The drive responds to a fault with an emergency stop ramp. |
| Fault | A fault is pending, the drive is stopped and disabled. |

19.32.4.2 Commands in the Control Word

Bit assignment in the control word

| Bit | Name | Bit | Name |
|-----|---|-----|-----------------------|
| 0 | Switch on | 8 | Pause/halt |
| 1 | Disable Voltage | 9 | reserved |
| 2 | Quick Stop | 10 | reserved |
| 3 | Enable Operation | 11 | reserved |
| 4 | Operation mode specific | 12 | reserved |
| 5 | Operation mode specific | 13 | Manufacturer-specific |
| 6 | Operation mode specific | 14 | Manufacturer-specific |
| 7 | Reset Fault (only effective for faults) | 15 | Manufacturer-specific |

Commands in the control word

| Command | Bit 7 Fault Reset | Bit 3 Enable Operation | Bit 2 Quick Stop | Bit 1 Disable Voltage | Bit 0 Switch On | Transitions |
|-------------------|-------------------------|------------------------------|------------------------|-----------------------------|-----------------------|--------------|
| Shutdown | X | X | 1 | 1 | 0 | 2, 6, 8 |
| Switch on | X | X | 1 | 1 | 1 | 3 |
| Disable Voltage | X | X | X | 0 | X | 7, 9, 10, 12 |
| Quick Stop | X | X | 0 | 1 | X | 7, 10, 11 |
| Disable Operation | X | 0 | 1 | 1 | 1 | 5 |
| Enable Operation | X | 1 | 1 | 1 | 1 | 4, 16 |
| Fault Reset | 1 | X | X | X | X | 15 |

Bits labeled **X** are irrelevant. **0** and **1** indicate the status of individual bits.

Mode-dependent bits in the control word

The following table shows the mode-dependent bits in the control word. Only manufacturer-specific modes are supported at present. The individual modes are set by Object 6060h Modes of operation.

| Operation mode | No | Bit 4 | Bit 5 | Bit 6 |
|----------------------------------|-----|------------------------|------------------------|-------------------|
| Profile Position Mode (pp) | 01h | new_setpoint | change_set_immediately | absolute/relative |
| Profile Velocity Mode (pv) | 03h | reserved | reserved | reserved |
| Profile Torque Mode (tq) | 04h | reserved | reserved | reserved |
| Homing Mode (hm) | 06h | homing_operation_start | reserved | reserved |
| Interpolated Position Mode (ip) | 07h | | reserved | reserved |
| Cyclic synchronous position mode | 08h | reserved | reserved | reserved |

Description of the remaining bits in the control word

Bit 8: (Pause) If Bit 8 is set, then the drive halts (pauses) in all modes. The setpoints (speed for homing or jogging, motion task number, setpoints for digital mode) for the individual modes are retained.

Bit 9,10: These bits are reserved for the drive profile (DS402).

Bit 13, 14, 15: These bits are manufacturer-specific, and reserved at present.

19.32.4.3 State Machine Bits (status word)

Bit assignment in the status word

| Bit | Name | Bit | Name |
|-----|--------------------|-----|------------------------------------|
| 0 | Ready to switch on | 8 | Manufacturer-specific (reserved) |
| 1 | Switched on | 9 | Remote (always 1) |
| 2 | Operation enable | 10 | Target reached |
| 3 | Fault | 11 | Internal limit active |
| 4 | Voltage enabled | 12 | Operation mode specific (reserved) |
| 5 | Quick stop | 13 | Operation mode specific (reserved) |
| 6 | Switch on disabled | 14 | Manufacturer-specific (reserved) |
| 7 | Warning | 15 | Manufacturer-specific (reserved) |

States of the state machine

| State | Bit 6 switch on disable | Bit 5 quick stop | Bit 3 fault | Bit 2 operation enable | Bit 1 switched on | Bit 0 ready to switch on |
|------------------------|----------------------------------|------------------------|----------------|------------------------------|-------------------------|-----------------------------------|
| Not ready to switch on | 0 | X | 0 | 0 | 0 | 0 |
| Switch on disabled | 1 | X | 0 | 0 | 0 | 0 |
| Ready to switch on | 0 | 1 | 0 | 0 | 0 | 1 |
| Switched on | 0 | 1 | 0 | 0 | 1 | 1 |
| Operation enabled | 0 | 1 | 0 | 1 | 1 | 1 |
| Fault | 0 | X | 1 | 0 | 0 | 0 |
| Fault reaction active | 0 | X | 1 | 1 | 1 | 1 |
| Quick stop active | 0 | 0 | 0 | 1 | 1 | 1 |

Bits labeled **X** are irrelevant. **0** and **1** indicate the status of individual bits.

Description of the remaining bits in the status word

Bit 4: `voltage_enabled` The DC-link voltage is present if this bit is set.

Bit 7: `warning` There are several possible reasons for Bit 7 being set and this warning being produced. The reason for this warning can be revealed by using the Object 20subindex manufacturer warnings.

Bit 9: `remote` is always set to 1, i.e. the drive can always communicate and be influenced via the RS232 - interface.

Bit 10: `target_reached` This is set when the drive has reached the target position.

Bit 11: `internal_limit_active` This bit specifies that a movement was or is limited. In different modes, different warnings cause the bit to be set.

19.32.5 Fixed PDO Mappings

Various ready-to-use mappings can be selected for cyclic data exchange via SDO's of the object 0x1C12 and 0x1C13. Using object 0x1C12 subindex 1 (Sync Manager 2 assignment), a fixed mapping for the cyclic command values can be set with the values 0x1701, 0x1702, 0x1720 to 0x1724. Using object 0x1C13 subindex 1 (Sync Manager 3 assignment), a fixed mapping for the cyclic actual values can be set via the data 0x1B01, 0x1B20 to 0x1B25.

Use the sequence below to select the fixed command value mapping 0x1701 via SDO's:

1. SDO write access to object 0x1C12Sub0 Data:0x00
2. SDO write access to object 0x1C12Sub1 Data:0x1701
3. SDO write access to object 0x1C12Sub0 Data:0x01

NOTE

Up to firmware version 1.8.x.x AKD.XML file, fixed mapping 0x1701 called out 0x6062sub0 as the "Position Command". From AKD PDMM firmware release 1.8.5.0, the AKD.XML will be changed to call out 0x60C1sub1 as the "Position Command" and an additional XML file called "AKD_TwinCAT.XML" will be added to support TwinCat 2x and older. In reality, SDO 0x6062sub0 is not supported in the AKD PDMM firmware but was called in the fixed mapping to support a TwinCat issue.

Position interface, supported fixed mappings:

| | |
|--------|--|
| 0x1701 | Position command value (4 bytes), Control word (2 bytes), total (6 bytes) |
| 0x1720 | Control Word (2 bytes), Interpolated position command value (4 bytes), Latch control word (2 bytes), Torque feed forward (2 bytes), Digital outputs (2 bytes) |
| 0x1721 | Interpolated position command value (4 bytes), Control Word (2 bytes), Torque feed forward (2 bytes) |
| 0x1722 | Control word (2 byte), Interpolated position command value (4 bytes), Latch control word (2 bytes), Torque feed forward (2 bytes), Digital outputs (2 bytes), max. torque (2 bytes) |
| 0x1723 | Control word (2 bytes), Interpolated position command value (4 bytes), Latch control word (2 bytes), Torque feed forward (2 bytes), Digital outputs (2 bytes), Reset of changed input information (2 bytes) |
| 0x1724 | Target position for cyclic synchronous position mode (4 bytes), Control word (2 byte), Torque feed forward (2 bytes) |
| 0x1B01 | Position actual value (4 bytes), Status word (2 bytes), total (6 bytes) |
| 0x1B20 | Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), velocity actual value (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position positive (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes) |
| 0x1B21 | Position Actual Internal Value (4 bytes), Status word (2 bytes) |
| 0x1B22 | Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), velocity actual value (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position negative (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes) |
| 0x1B23 | Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), velocity actual value (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position positive / negative (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes) |
| 0x1B24 | Position actual value (4 bytes), status word (2 bytes) |

| | |
|--------|---|
| 0x1B25 | Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), latch position 2 positive / negative (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position 1 positive / negative (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes) |
|--------|---|

Velocity interface, supported fixed mappings:

| | |
|--------|---|
| 0x1702 | Velocity command value (4 bytes), Control word (2 bytes), total (6 bytes) |
|--------|---|

The objects, which are mapped into the fixed PDOs can be read via the subindices 1 to n of the above indices. The number of mapped entries is available by reading subindex 0 of the above indices.

Example:

A read access to object 1702 sub 0 gives a value of 2, a read on subindex 1 gives 0x60ff0020, on subindex 2 0x60400010. The meaning of these numbers can be seen in the CANopen manual or the flexible-mapping example (→ p. 948.).

19.32.6 Flexible PDO Mappings

In addition to the fixed PDO mapping the so-called flexible mapping of real-time objects is possible.

NOTE

Available objects for PDO mapping are listed in the object dictionaries ("Appendix" (→ p. 968))("Appendix" (→ p. 1)). All objects with the entry "yes" in column "PDO map." can be used.

Restrictions of flexible mapping:

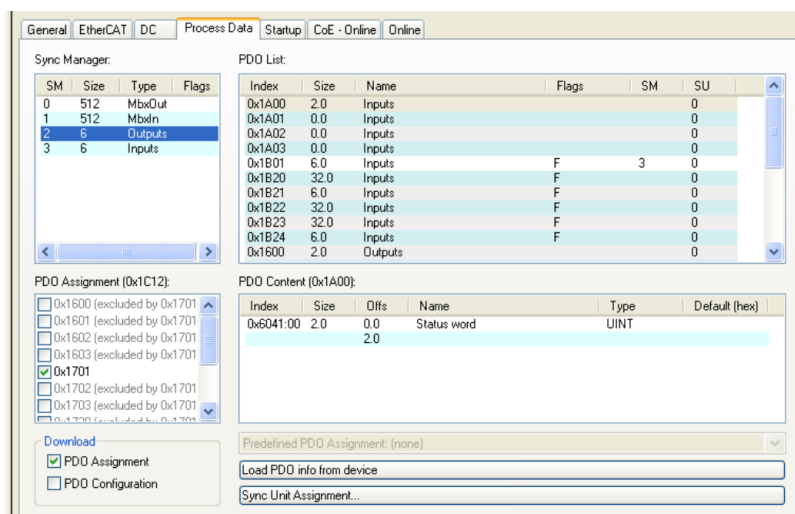
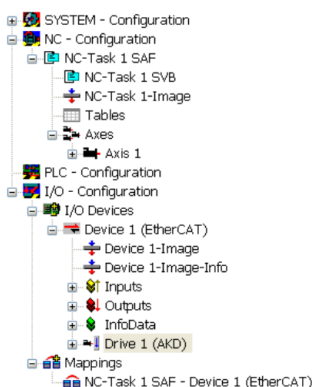
- An odd length PDO is not allowed.
- In the Rx(=set-point)-direction the dummy-object 0x0002 sub 0 with a length of 8 bits can be used to make the PDO-length even.
- In the Tx(=actual value)-direction one sub-index of the manufacturer status object 0x2002 sub 1..4 can be used to guarantee the even length of the Tx-PDO.
- These special mappings may be used if the objects 0x6060 and 0x6061 have to be used in the mapping.
- The allowed PDOs have up to 32 bytes (Tx) or 22 bytes (Rx). They are built from smaller PDO modules with a maximum length of 8 bytes. These are built by using the mapping objects 0x1600 to 0x1603 and 0x1a00 to 0x1a03.

The configuration is similar to the described sequence for the fixed mappings:

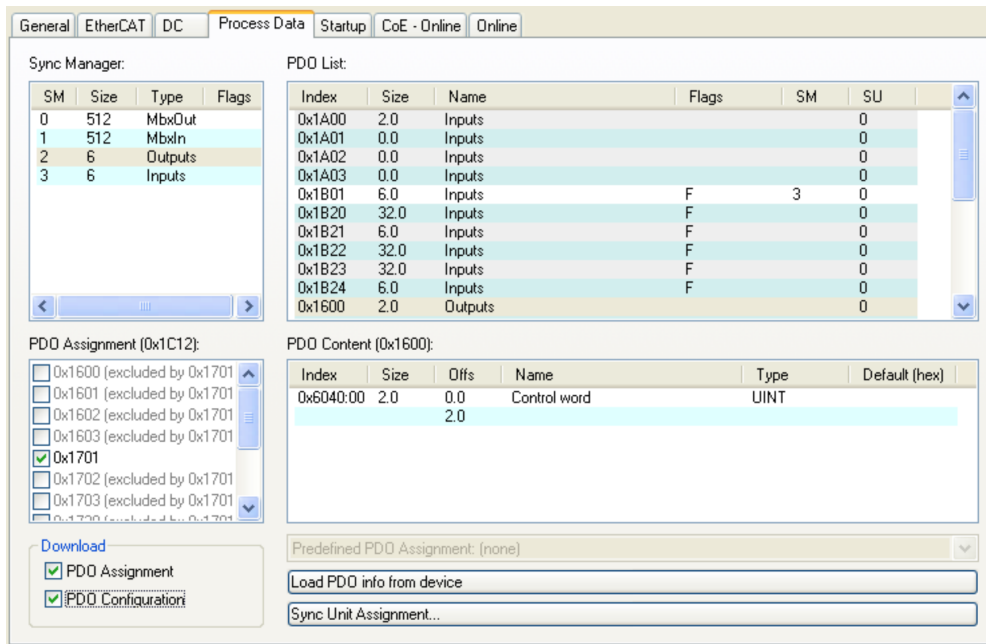
1. The mapping selection is cleared (write 0 to object 0x1C12 sub 0 and 1C13 sub 0)
2. As the AKD PDMM - implementation is based on CANopen the real-time data are build from up to 4 PDOs with 8 bytes in both directions. These PDOs are built in the same way as in a CAN-drive with the objects 0x1600 - 0x1603 and 0x1A00 - 0x1A03. Unused PDOs must be cleared with writing 0 to the subindex 0.
3. SDO write access to object 0x1C12 sub 1 .. 4 with the PDOs (0x1600 .. 0x1603), that should be used in receive direction of the AKD PDMM (set point values).
4. SDO write access to object 0x1C13 sub 1 .. 4 with the PDOs (0x1A00 .. 0x1A03), that should be used in transmit direction of the AKD PDMM (actual values).
5. SDO write access to the objects 0x1C12 sub 0 and 0x1C13 sub 0 with the number of mapped PDOs in this direction.

See an example in chapter "Flexible PDO Mappings" (→ p. 791).

The cyclically used data are visible in the PDO-assignment window for the Inputs and Outputs of the Sync Managers. Default setting are the fixed PDOs 0x1701 and 0x1B01 (visible contents when selected in the PDO list).



If the flexible mapping is required, the PDO configuration check box must be changed.

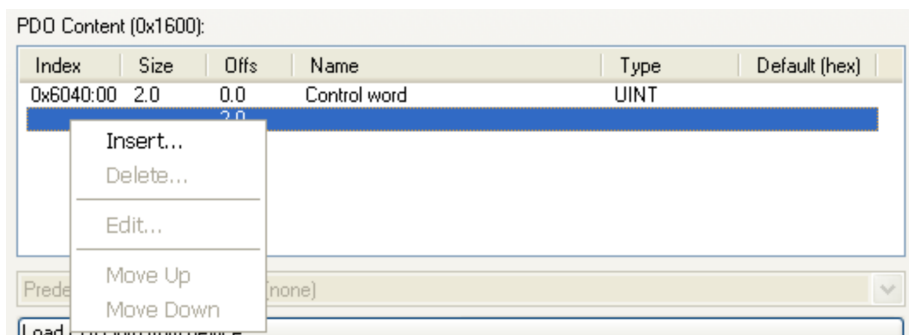


19.32.6.1 Example: Flexible PDO Mapping

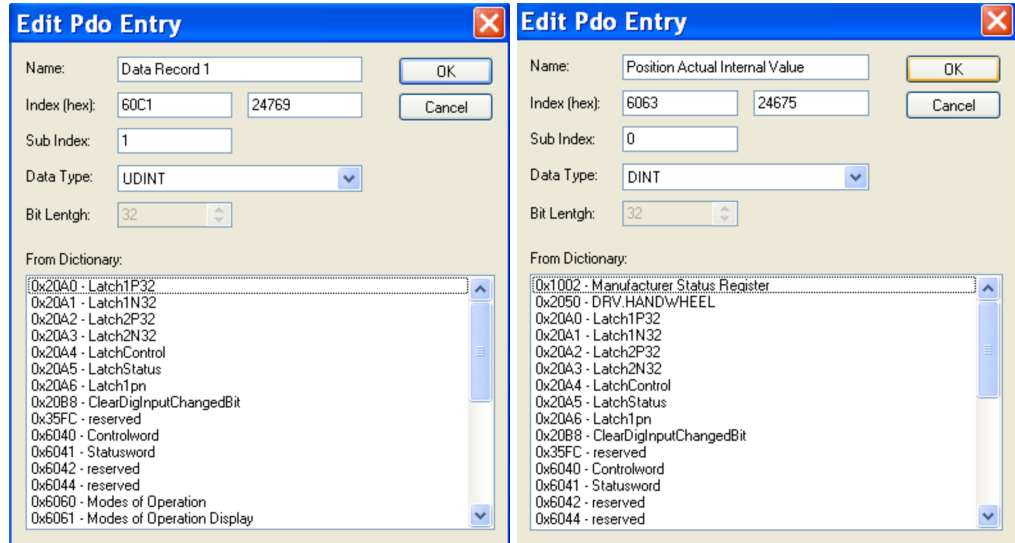
For the flexible mapping of the Outputs the fixed mapping 0x1701 has to be switched off and up to 4 free-mappable PDOs (0x1600-0x1603) can be used instead. The maximum number of bytes for each of these PDOs is 8.



After that the default mapping of e.g. the PDO 0x1600 can be extended:



A list of possible objects for the mapping will be shown and a new entry can be chosen.



In this case the setpoint for the interpolated position mode is selected.

The same is valid for the Tx-PDO-direction. Here the value of the actual internal position is selected.

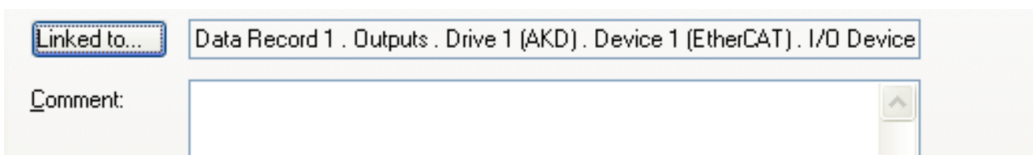
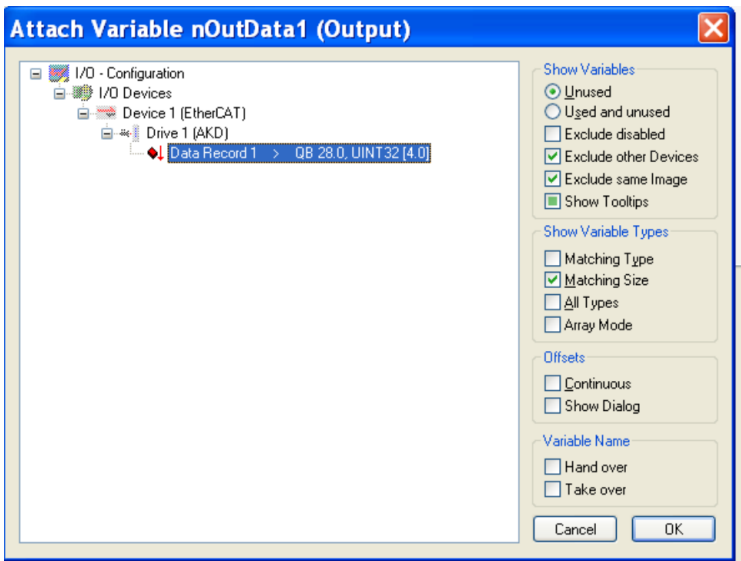
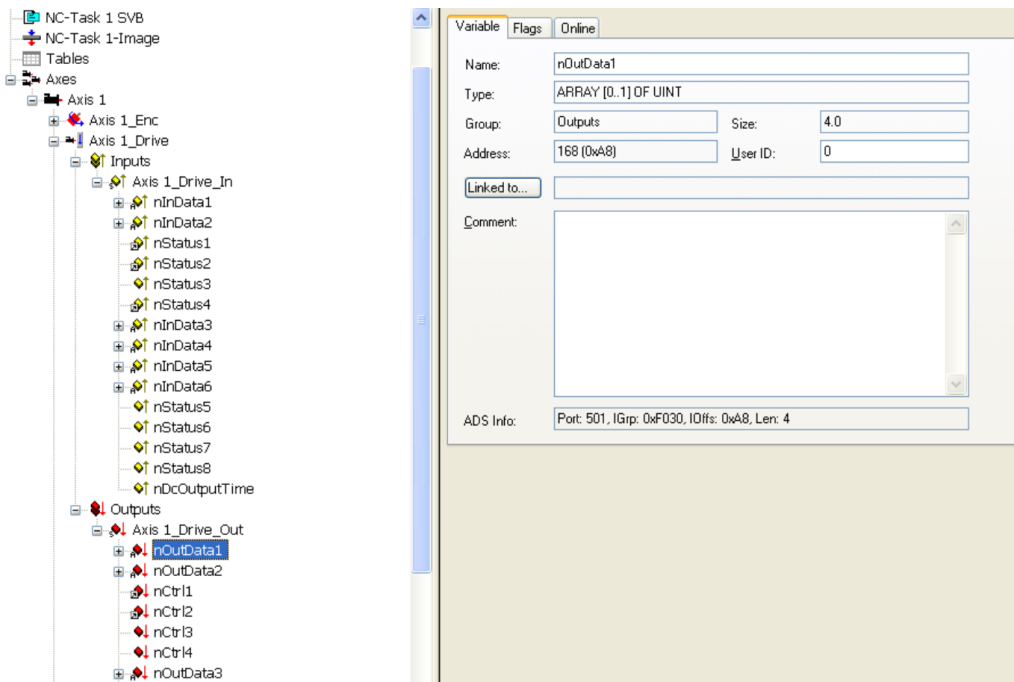
This results in the startup-SDO-list for this sample free-mapped-configuration.

| Transition | Protocol | Index | Data | Comment |
|------------|----------|-----------|-------------------------|-----------------------------|
| <PS> | CoE | 0x1C12:00 | 0x00 (0) | clear sm pdos (0x1C12) |
| <PS> | CoE | 0x1C13:00 | 0x00 (0) | clear sm pdos (0x1C13) |
| <PS> | CoE | 0x1A00:00 | 0x00 (0) | clear pdo 0x1A00 entries |
| <PS> | CoE | 0x1A00:01 | 0x60410010 (1614872592) | download pdo 0x1A00 entry |
| <PS> | CoE | 0x1A00:02 | 0x60630020 (1617100832) | download pdo 0x1A00 entry |
| <PS> | CoE | 0x1A00:00 | 0x02 (2) | download pdo 0x1A00 entr... |
| <PS> | CoE | 0x1A01:00 | 0x00 (0) | clear pdo 0x1A01 entries |
| <PS> | CoE | 0x1A02:00 | 0x00 (0) | clear pdo 0x1A02 entries |
| <PS> | CoE | 0x1A03:00 | 0x00 (0) | clear pdo 0x1A03 entries |
| <PS> | CoE | 0x1600:00 | 0x00 (0) | clear pdo 0x1600 entries |
| <PS> | CoE | 0x1600:01 | 0x60400010 (1614807056) | download pdo 0x1600 entry |
| <PS> | CoE | 0x1600:02 | 0x60C10120 (1623261472) | download pdo 0x1600 entry |
| <PS> | CoE | 0x1600:00 | 0x02 (2) | download pdo 0x1600 entr... |
| <PS> | CoE | 0x1601:00 | 0x00 (0) | clear pdo 0x1601 entries |
| <PS> | CoE | 0x1602:00 | 0x00 (0) | clear pdo 0x1602 entries |
| <PS> | CoE | 0x1603:00 | 0x00 (0) | clear pdo 0x1603 entries |
| <PS> | CoE | 0x1C12:01 | 0x1600 (5632) | download pdo 0x1C12:01 i... |
| <PS> | CoE | 0x1C12:00 | 0x01 (1) | download pdo 0x1C12 count |
| <PS> | CoE | 0x1C13:01 | 0x1B01 (6913) | download pdo 0x1C13:01 i... |
| <PS> | CoE | 0x1C13:00 | 0x01 (1) | download pdo 0x1C13 count |
| PS | CoE | 0x6060:00 | 0x07 (7) | Op mode |
| PS | CoE | 0x60C2:01 | 0x02 (2) | Cycle time |
| PS | CoE | 0x60C2:02 | 0xFD (253) | Cycle exp |

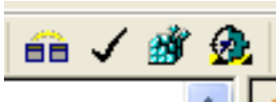
The meaning of the data (for example 0x60410010 in the mapping of 0x1A00 sub 1) is as follows:

- 0x6041 is the index of the DS402 status word
- 0x00 is the subindex of the DS402 status word
- 0x10 is the number of bits for this entry, i. e. 16 bits or 2 bytes.

If this shall be used in the NC, the interpolation set point position has to be linked from the axis to the NC-axis.



After doing this configuration the mapping can be activated as seen before in this document:



Now the NC-screen should show a position in the online window, which changes a bit in the last digits.

| General | Settings | Parameter | Dynamics | Online | Functions | Coupling | Compensation |
|---|----------|---|----------|---|-------------------------------------|----------------|--------------|
| | | | | 165.7124 | Setpoint Position: [mm] 165.5651 | | |
| Lag Distance (min/max): [mm] 0.0010 (-0.025, 0.002) | | Actual Velocity: [mm/s] -0.0168 | | Setpoint Velocity: [mm/s] 0.0000 | | Error: 0 (0x0) | |
| Override: [%] 100.0000 % | | Total / Control Output: [%] 0.00 / 0.00 % | | | | | |
| Status (log.) <input type="checkbox"/> Ready <input type="checkbox"/> Calibrated <input type="checkbox"/> Has Job <input checked="" type="checkbox"/> NDT Moving <input type="checkbox"/> Moving Fw <input type="checkbox"/> Moving Bw | | Status (phys.) <input type="checkbox"/> Coupled Mode <input type="checkbox"/> In Target Pos. <input type="checkbox"/> In Pos. Range | | Enabling <input type="checkbox"/> Controller <input type="button" value="Set"/> <input type="checkbox"/> Feed Fw <input type="checkbox"/> Feed Bw | | | |
| Controller K _v -Factor: [mm/s/mm] 1 | | Reference Velocity: [mm/s] 2200 | | | | | |
| Target Position: [mm] 0 | | Target Velocity: [mm/s] 0 | | | | | |
| <input type="button" value="--"/> F1 <input type="button" value="-"/> F2 <input type="button" value="+"/> F3 <input type="button" value="++"/> F4 | | <input type="button" value="◇"/> F5 <input type="button" value="⊖"/> F6 | | <input type="button" value="Ⓜ"/> F8 <input type="button" value="→•"/> F9 | | | |

After enabling the power stage with the All-button, the drive can be moved via the jog-buttons or via the functions in the function menu.

Set Enabling ✕

Controller
 Feed Fw
 Feed Bw

Override [%]:

F3

F4

19.32.7 Supported Cyclical Setpoint and Actual Values

Supported cyclical setpoint values

| Name | CANopen object | Data type | Description |
|----------------------------------|----------------|-----------|--|
| Target current | 0x2071 sub 0 | 32 bit | scaled in mA |
| Latch Control word | 0x20A4 sub 0 | UINT16 | |
| Clear digital Input Change Bit | 0x20B8 | 16 bit | |
| Analog output value | 0x3470 sub 3 | 16 bit | |
| External feedback position | 0x3497 sub 0 | 32 bit | |
| CANopen control-word | 0x6040 sub 0 | UINT16 | CANopen control word. |
| Modes of Operation | 0x6060 sub 0 | 8 bit | DS402 opmode setpoint |
| Velocity Window | 0x606D sub 0 | 16 bit | |
| Velocity Window Time | 0x606E sub 0 | 16 bit | |
| Target Torque | 0x6071 sub 0 | 16 bit | 0.1% resolution |
| Maximum Torque | 0x6072 sub 0 | 16 bit | |
| Target position | 0x607A sub 0 | INT32 | Used in profile position mode / cyclic synchronous position mode |
| Profile position target velocity | 0x6081 sub 0 | 32 bit | related to MT.V |
| Profile position target acc | 0x6083 sub 0 | 32 bit | related to MT.ACC |
| Profile position target dec | 0x6084 sub 0 | 32 bit | related to MT.DEC |
| Velocity feed forward | 0x60B1 sub 0 | 32 bit | |
| Torque feed forward | 0x60B2 sub 0 | INT16 | |
| Touch probe function | 0x60B8 | 16 bit | |
| Position command value | 0x60C1 sub 1 | INT32 | Interpolation data record in IP-mode |
| Digital outputs | 0x60FE sub 1 | UINT32 | |
| Velocity command value | 0x60FF sub 0 | INT32 | |

Supported cyclical actual values

| Name | CANopen object | Data type | Description |
|---------------------------------|----------------|-----------|-----------------------|
| Position actual internal value | 0x6063 sub 0 | INT32 | |
| Velocity actual value | 0x606C sub 0 | INT32 | |
| CANopen status-word | 0x6041 sub 0 | UINT16 | CANopen status word |
| Second position feedback | 0x2050 sub 0 | INT32 | |
| Digital inputs | 0x60FD sub 0 | UINT32 | |
| Following error actual value | 0x60F4 sub 0 | INT32 | |
| Latch position positive edge | 0x20A0 sub 0 | INT32 | |
| Torque actual value | 0x6077 sub 0 | INT16 | |
| Latch status | 0x20A5 sub 0 | UINT16 | |
| Actual Current | 0x2077 sub 0 | 32 bit | scaled in mA |
| Latch1 negative edge | 0x20A1 sub 0 | 32 bit | |
| Latch2 Positive | 0x20A2 sub 0 | 32 bit | |
| Latch2 Negative | 0x20A3 sub 0 | 32 bit | |
| Latch1 positive/negative edge | 0x20A6 | 32 bit | |
| Latch 2 positive/negative edge | 0x20A7 | 32 bit | |
| Modes of Operation | 0x6061 | 8 bit | DS402 opmode status |
| Position Actual Value | 0x6064 sub 0 | 32 bit | WB/ DS402 scale units |
| Touch probe status | 0x60B9 sub 0 | 16 bit | |
| Touch probe 1 positive edge pos | 0x60BA sub 0 | 32 bit | |
| Touch probe 1 negative edge pos | 0x60BB sub 0 | 32 bit | |
| Touch probe 2 positive edge pos | 0x60BC sub 0 | 32 bit | |
| Touch probe 2 negative edge pos | 0x60BD sub 0 | 32 bit | |
| Additional Pos actual value | 0x60E4 sub 0 | 48 bit | |
| Additional Pos actual value | 0x60E4 sub 1 | 32 bit | |
| Motor I2t | 0x3427 sub 3 | 32 bit | |
| Analog output value | 0x3470 sub 2 | 16 bit | |
| Analog Input value | 0x3470 sub 4 | 16 bit | |
| Manufacturer status register | 0x1002 sub 0 | 32 bit | |

19.32.8 Supported Operation Modes

| CANopen mode of operation | AKD PDMM mode of operation | Description |
|-----------------------------|---------------------------------|--|
| Profile velocity | DRV.OPMODE 1 DRV.CMDSOURCE 1 | 0x6060Sub0 Data: 3 In this mode, the EtherCAT master sends cyclic velocity command values to the AKD PDMM. |
| Interpolated position | DRV.OPMODE 2 DRV.CMDSOURCE 1 | 0x6060Sub0 Data: 7 In this mode of operation the EtherCAT master sends cyclic position command values to the AKD PDMM. These command values are interpolated by the AKD PDMM according to the fieldbus sample rate. |
| Homing mode | DRV.OPMODE 2 DRV.CMDSOURCE 0 | 0x6060 sub 0 data : 6 In this mode an AKD PDMM-internal homing can be done. |
| Profile Position | DRV.OPMODE 2 DRV.CMDSOURCE 0 | 0x6060sub0 Data: 1 Uses motion task 0 to execute a point to point move |
| Torque | DRV.OPMODE 0 DRV.CMDSOURCE 1 | 0x6060sub0 Data: 4 Commands torque in % of drive peak torque |
| Cyclic Synchronous Position | DRV.OPMODE 2 DRV.CMDSOURCE 1 | 0x6060sub0 Data: 8 Master calculates move profile and commands motion with position points |

19.32.9 Adjusting EtherCAT Cycle Time

The cycle time to be used in the drive for the cyclical setpoints and actual values can either be stored in the FBUS.SAMPLEPERIOD parameter in the amplifier or configured in the startup phase. This happens via SDO mailbox access to objects 60C2 subindex 1 and 2.

Subindex 2, known as the interpolation time index, defines the power of ten of the time value (e.g. -3 means 10⁻³ or milliseconds) while subindex 1, known as interpolation time units, gives the number of units (e.g. 4 means 4 units).

You can run a 2 ms cycle using various combinations. For example,

Index = -3, Units = 2 or

Index = -4, Units = 20 etc.

The FBUS.SAMPLEPERIOD parameter is counted in multiples of 62.5us microseconds within the device. This means, for example that 2 ms equates to FBUS.SAMPLEPERIOD value of 32.

19.32.10 Maximum Cycle Times depending on operation mode

The minimum cycle time for the drive is largely dependent on the drive configuration (second actual position value encoder latch functionality enabled and so on)

| Interface | Cycle time AKD PDMM |
|-----------|----------------------|
| Position | ≥ 0.25 ms (≥ 250 μs) |
| Velocity | ≥ 0.25 ms (≥ 250 μs) |
| Torque | ≥ 0.25 ms (≥ 250 μs) |

19.32.11 Synchronization

On all drives, the internal PLL is theoretically able to even out an average deviation of up to 4800 ppm in the cycle time provided by the master. The drive checks once per fieldbus cycle a counter within the drive internal FPGA, which is cleared by a Sync0 (Distributed clock) event. Depending of the counter value, the drive extends or decreases the 62.5 µs MTS signal within the drive by a maximum of 300 ns.

The theoretical maximum allowed deviation can be calculated by using the following formula:

$$\max_{dev} = \frac{300[\text{ns}]}{62.5[\mu\text{s}]} \cdot 1,000,000 = 4800 [\text{ppm}]$$

The synchronization functionality within the drive can be enabled via setting bit 0 of the FBUS.PARAM02 parameter to high. Therefore FBUS.PARAM02 must be set to the value of 1. Furthermore the distributed clock functionality must be enabled by the EtherCAT master in order to activate cyclic Sync0 events.

19.32.11.1 Synchronization behavior with distributed clocks (DC) enabled

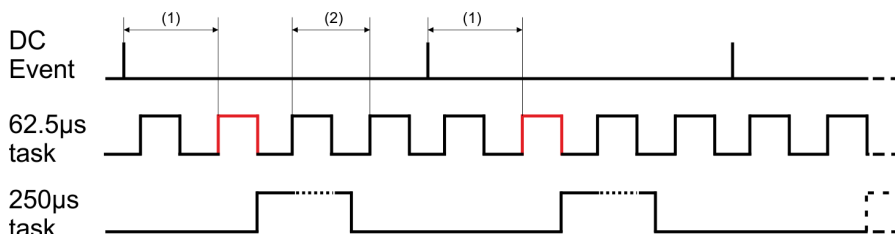
When the EtherCAT master enables distributed clocks, a distributed clock (DC) event is created in the AKD PDMM once per fieldbus cycle. An assigned 62.5 µs real-time task in the AKD PDMM monitors the elapsed time between the DC events and the AKD PDMM System time and extends or reduces the 62.5 µs strobe to the CPU as necessary.

The following fieldbus parameters are used for the synchronization feature:

1. FBUS.SYNCDIST = Expected time delay of the AKD PDMM PLL-code to the DC event.
2. FBUS.SYNCACT = Actual time delay of the AKD PDMM PLL-code to the DC event.
3. FBUS.PLLTHRESH = Number of consecutive successful synchronized PLL cycles of the AKD PDMM before the Drive is considered as synchronized.
4. FBUS.SYNCWND = Synchronization window in which the AKD PDMM is considered to be synchronized. The Drive is considered synchronized as long as the following statement is true for FBUS.PLLTHRESH consecutive cycles:

$$\text{FBUS.SYNCDIST} - \text{FBUS.SYNCWND} < \text{FBUS.SYNCACT} < \text{FBUS.SYNCDIST} + \text{FBUS.SYNCWND}$$

Example with a 4kHz fieldbus sample rate:



Explanation: The red-marked 62.5[µs] real-time task displays the AKD PDMM 62.5 µs real-time task within one fieldbus cycle which is responsible for calling the AKD PDMM PLL-code. The time delay (1) shows the actual delay to the previous DC event, which is ideally close to the adjusted FBUS.SYNCDIST parameter. Depending on (1) the AKD PDMM slightly extends or reduce the 62.5[µs] IRQ generation of the high-priority real-time task in order to either increase or decrease the measured time delay to the DC event (1) for the next PLL cycle. The time distance (2) shows the 62.5[µs] ± x[ms] realtime task of the AKD PDMM.

19.32.11.2 Synchronization behavior with distributed clocks (DC) disabled

The AKD PDMM fieldbus synchronization algorithm is similar to that used by Distributed Clocks. The difference is that the AKD PDMM synchronizes to a SyncManager2 event instead of the DC event. A SyncManager2 event is created when the EtherCAT Master sends a new package of command values to the drive while the network is in the Operational state. This occurs once per fieldbus cycle.

19.32.12 Latch Control Word and Latch Status Word

Latch Control word (2 Byte)

| Bit | Value (bin) | Value (hex) | Description |
|-------|----------------------|-------------|---------------------------------------|
| 0 | 00000000 00000001 | zz01 | Enable extern latch 1 (positive rise) |
| 1 | 00000000 00000010 | zz02 | Enable extern latch 1 (negative rise) |
| 2 | 00000000 00000100 | zz04 | Enable extern latch 2 (positive rise) |
| 3 | 00000000 00001000 | zz08 | Enable extern latch 2 (negative rise) |
| 4 | | | |
| 5-7 | | | Reserve |
| 8-12 | 00000001 00000000 | 01zz | Read external latch 1 (positive rise) |
| | 00000010 00000000 | 02zz | Read external latch 1 (negative rise) |
| | 00000011 00000000 | 03zz | Read external latch 2 (positive rise) |
| | 00000100 00000000 | 04zz | Read external latch 2 (negative rise) |
| 13-15 | | | Reserve |

Latch Status word (2 Byte)

| Bit | Value (bin) | Value (hex) | Description |
|-------|-------------------|-------------|--|
| 0 | 00000000 00000001 | zz01 | External latch 1 valid (positive rise) |
| 1 | 00000000 00000010 | zz02 | External latch 1 valid (negative rise) |
| 2 | 00000000 00000100 | zz04 | External latch 2 valid (positive rise) |
| 3 | 00000000 00001000 | zz08 | External latch 2 valid (negative rise) |
| 4 | | | |
| 5-7 | | | Reserve |
| 8-11 | 00000001 00000000 | z1zz | Acknowledge value external latch 1 (positive rise) |
| | 00000010 00000000 | z2zz | Acknowledge value external latch 1 (negative rise) |
| | 00000011 00000000 | z3zz | Acknowledge value external latch 2 (positive rise) |
| | 00000100 00000000 | z4zz | Acknowledge value external latch 2 (negative rise) |
| 12-15 | 00010000 00000000 | 1zzz | Zustand Digital Input 4 |
| | 00100000 00000000 | 2zzz | Zustand Digital Input 3 |
| | 01000000 00000000 | 4zzz | Zustand Digital Input 2 |

| Bit | Value (bin) | Value (hex) | Description |
|-----|-------------------|-------------|-------------------------|
| | 10000000 00000000 | 8zzz | Zustand Digital Input 1 |

19.32.13 Mailbox Handling

With EtherCAT, acyclical data traffic (object channel or SDO channel) is called mailbox.

NOTE

Available SDO objects are listed in the ("Appendix" (→ p. 968))("Appendix" (→ p. 1)).

This system is based around the master:

Mailbox Output:

The master (EtherCAT controller) sends data to the slave (drive). This is essentially a (read/write) request from the master. Mailbox output operates via Sync Manager 0.

Mailbox Input:

The slave (drive) sends data to the master (EtherCAT controller). The master reads the slave's response. Mailbox input operates via Sync Manager 1.

Timing diagram

The timing diagram illustrates the mailbox access process:



1. The EtherCAT master writes the mailbox request to the mail-out buffer.
2. On the next interrupt, the EtherCAT interface activates a Sync Manager 0 event (mailbox output event) in the AL event register.
3. The drive reads 16 bytes from the mail-out buffer and copies them to the internal mailbox output array.
4. The drive identifies new data in the internal mailbox output array and performs an SDO access to the object requested by the EtherCAT interface. The response from the drive is written to an internal mailbox input array.
5. The drive deletes all data in the internal mailbox output array so that a new mailbox access attempt can be made.
6. The drive copies the response telegram from the internal mailbox input array to the mail-in buffer of the EtherCAT interface.

19.32.13.1 Mailbox Output

An interrupt by the EtherCAT-interface with a Sync Manager 0 - Event starts a Mailbox Output Process. A 1 in the Mail Out Event-Bit of the AL Event register signals the drive, that the EtherCAT-interface wants to send a Mailbox message and that it has already stored the required data in the Mail Out Buffer. Now 16 Byte data are read by the drive with the IRQ process. The bytes are defined as follows

| Address 0x1800 | | | | | | | | Address 0x180F | | | | | | | |
|---|---|---|---|---|---|---|------------------|--|---|----|----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| CAN over EtherCAT specific data (CoE Header) | | | | | | | | CAN specific data (standard CAN SDO) | | | | | | | |
| Byte 0 | Length of the data (Low Byte) | | | | | | | | | | | | | | |
| Byte 1 | Length of the data (High Byte) | | | | | | | | | | | | | | |
| Byte 2 | Address (Low Byte) | | | | | | | | | | | | | | |
| Byte 3 | Address (High Byte) | | | | | | | | | | | | | | |
| Byte 4 | Bit 0 to 5: Channel Bit 6 to 7: Priority | | | | | | | | | | | | | | |
| Byte 5 | Bit 0 to 3: Type | | | | | | | 1 = Reserved: ADS over EtherCAT 2 = Reserved: Ethernet over EtherCAT 3 = Can over EtherCAT...) | | | | | | | |
| | Bit 4 to 7: Reserved | | | | | | | | | | | | | | |
| Byte 6 | PDO Number (with PDO transmissions only, Bit 0 = LSB of the PDO number, see Byte 7 for MSB) | | | | | | | | | | | | | | |
| Byte 7 | Bit 0: MSB of the PDO number, see Byte 6 | | | | | | | | | | | | | | |
| | Bit 1 to 3: Reserved | | | | | | | | | | | | | | |
| | Bit 4 to 7: CoE specific type | | | | | | | 0: Reserved | | | | | | | |
| | | | | | | | | 1: Emergency message | | | | | | | |
| | | | | | | | | 2: SDO request | | | | | | | |
| | | | | | | | | 3: SDO answer | | | | | | | |
| | | | | | | | | 4: TxPDO | | | | | | | |
| | | | | | | | | 5: RxPDO | | | | | | | |
| | | | | | | | | 6: Remote transmission request of a TxPDO | | | | | | | |
| | | | | | | | | 7: Remote transmission request of a RxPDO | | | | | | | |
| | | | | | | | 8...15: reserved | | | | | | | | |
| Byte 8 | Control-Byte in the CAN telegram: | | | | | | | | | | | | | | |
| | write access: | | | | | | | 0x23=4Byte, 0x27=3Byte, 0x2B=2Byte, 0x2F-F=1Byte | | | | | | | |
| | read access: | | | | | | | 0x40 | | | | | | | |
| Byte 9 | Low Byte of the CAN object number (Index) | | | | | | | | | | | | | | |
| Byte 10 | High Byte of the CAN object number (Index) | | | | | | | | | | | | | | |
| Byte 11 | Subindex according to CANopen Specification for the drive | | | | | | | | | | | | | | |
| Byte 12 | Data with a write access (Low Byte) | | | | | | | | | | | | | | |
| Byte 13 | Data with a write access | | | | | | | | | | | | | | |
| Byte 14 | Data with a write access | | | | | | | | | | | | | | |

| | |
|---------|--------------------------------------|
| Byte 15 | Data with a write access (High Byte) |
|---------|--------------------------------------|

The drive answers every telegram with an answer in the Mailbox Input buffer.

19.32.13.2 Mailbox Input

The drive answers every CoE telegram with a 16 byte answer telegram in the Mailbox Input buffer. The bytes are defined as follows:

| Address 0x1C00 | | | | | | | | Address 0x1C0F | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | |
| CAN over EtherCAT specific data (CoE Header) | | | | | | | | CAN specific data (standard CAN SDO) | | | | | | | | |
| Byte 0 | Length of the data (Low Byte) | | | | | | | | | | | | | | | |
| Byte 1 | Length of the data (High Byte) | | | | | | | | | | | | | | | |
| Byte 2 | Address (Low Byte) | | | | | | | | | | | | | | | |
| Byte 3 | Address (High Byte) | | | | | | | | | | | | | | | |
| Byte 4 | Bit 0 to 5: Channel Bit 6 to 7: Priority | | | | | | | | | | | | | | | |
| Byte 5 | Bit 0 to 3: Type | | | | | | | 1 = Reserved: ADS over EtherCAT 2 = Reserved: Ethernet over EtherCAT 3 = Can over EtherCAT... | | | | | | | | |
| | Bit 4 to 7: Reserved | | | | | | | | | | | | | | | |
| Byte 6 | PDO Number (with PDO transmissions only, Bit 0 = LSB of the PDO number, see Byte 7 for MSB) | | | | | | | | | | | | | | | |
| Byte 7 | Bit 0: MSB of the PDO number, see Byte 6 | | | | | | | | | | | | | | | |
| | Bit 1 to 3: Reserved | | | | | | | | | | | | | | | |
| | Bit 4 to 7: CoE specific type | | | | | | | 0: Reserved | | | | | | | | |
| | | | | | | | | 1: Emergency message | | | | | | | | |
| | | | | | | | | 2: SDO request | | | | | | | | |
| | | | | | | | | 3: SDO answer | | | | | | | | |
| | | | | | | | | 4: TXPDO | | | | | | | | |
| | | | | | | | | 5: RxPDO | | | | | | | | |
| | | | | | | | | 6: Remote transmission request of a TxPDO | | | | | | | | |
| | | | | | | | | 7: Remote transmission request of a RxPDO | | | | | | | | |
| Byte 8 | Control-Byte in the CAN telegram: | | | | | | | | | | | | | | | |
| | write access OK: | | | | | | | 0x60 | | | | | | | | |
| | read access OK + length of answer: | | | | | | | 0x43 (4 Byte), 0x47 (3 Byte), 0x4B (2Byte), 0x4F (1Byte) | | | | | | | | |
| | error with read- or write access: | | | | | | | 0x80 | | | | | | | | |
| Byte 9 | Low Byte of the CAN object number (Index) | | | | | | | | | | | | | | | |
| Byte 10 | High Byte of the CAN object number (Index) | | | | | | | | | | | | | | | |
| Byte 11 | Subindex according to CANopen Specification for Kollmorgen drive | | | | | | | | | | | | | | | |
| Byte 12 | Data (Low Byte) | | | | | | | | | | | | | | | |

| | | |
|---------|------------------|--|
| Byte 13 | Data | error code Fehlercode according to CANopen Specification in case of an error |
| Byte 14 | Data | data value of the object in case of successful read access |
| Byte 15 | Data (High Byte) | |

19.32.13.3 Example: Mailbox Access

In the example below, PDOs 0x1704 are mapped (see Chapter "Fixed PDO Mappings" (→ p. 943) "Fixed PDO Mappings"):

The master sends this mailbox output message:

| | | |
|---------|------|--|
| Byte 0 | 0x0A | The next 10 Bytes contain data (Byte 2 to Byte 11) |
| Byte 1 | 0x00 | The next 10 Bytes contain data (Byte 2 to Byte 11) |
| Byte 2 | 0x00 | Address 0 |
| Byte 3 | 0x00 | Address 0 |
| Byte 4 | 0x00 | Channel 0 and Priority 0 |
| Byte 5 | 0x03 | CoE Object |
| Byte 6 | 0x00 | PDO Number 0 |
| Byte 7 | 0x20 | PDO Number 0 and SDO-Request |
| Byte 8 | 0x2B | 2 Byte write access |
| Byte 9 | 0x12 | SDO-Object 0x1C12 |
| Byte 10 | 0x1C | SDO-Object 0x1C12 |
| Byte 11 | 0x01 | Subindex 1 |
| Byte 12 | 0x04 | Data value 0x00001704 |
| Byte 13 | 0x17 | Data value 0x00001704 |
| Byte 14 | 0x00 | Data value 0x00001704 |
| Byte 15 | 0x00 | Data value 0x00001704 |

The drive returns the following message:

| | | |
|---------|------|--|
| Byte 0 | 0x0E | The next 14 Bytes contain data (Byte 2 to Byte 15) |
| Byte 1 | 0x00 | The next 14 Bytes contain data (Byte 2 to Byte 15) |
| Byte 2 | 0x00 | Address 0 |
| Byte 3 | 0x00 | Address 0 |
| Byte 4 | 0x00 | Channel 0 and Priority 0 |
| Byte 5 | 0x03 | CoE Object |
| Byte 6 | 0x00 | PDO Number 0 |
| Byte 7 | 0x20 | PDO Number 0 and SDO-Answer |
| Byte 8 | 0x60 | Successful write access |
| Byte 9 | 0x12 | SDO-Object 0x1C12 |
| Byte 10 | 0x1C | SDO-Object 0x1C12 |
| Byte 11 | 0x01 | Subindex 1 |
| Byte 12 | 0x00 | Data value 0x00000000 |
| Byte 13 | 0x00 | Data value 0x00000000 |
| Byte 14 | 0x00 | Data value 0x00000000 |
| Byte 15 | 0x00 | Data value 0x00000000 |

19.32.14 EEPROM Content

AKD PDMM has a built-in emulated EEPROM. This EEPROM can be read by the EtherCAT master to get some information about drive properties, like PDO-information, drive name, serial numbers and communication-specific attributes.

They are organized in categories. There are two manufacturer-specific categories implemented in the AKD PDMM:

- Category 0x0800: Holds a string with the model type in the format AKD PDMM-P00000-NxxC-0000
- Category 0x0801: Holds the firmware version in the format 0x_xx-xx-yyy

This page intentionally left blank.

19.33 Appendix

19.33.1 CANopen Emergency Messages and Error Codes

Emergency messages are triggered by internal equipment errors. They have a high ID-priority to ensure quick access to the bus. An emergency message contains an error field with pre-defined error/fault numbers (2 bytes), an error register (1byte), the error category (1 byte), and additional information. Error numbers 0000h to 7FFFh are defined in the communication or drive profile. Error numbers FF00h to FFFFh have manufacturer-specific definitions.

| Error Code | Fault/Warning | Description |
|------------|---------------|--|
| 0x0000 | 0 | Emergency error free. |
| 0x1080 | - | General Warning. |
| 0x1081 | - | General Error. |
| 0x3110 | F523 | DC Bus link over voltage FPGA. |
| 0x3120 | F247 | DC Bus link voltage exceed allowed thresholds. |
| 0x3130 | F503 | DC Bus link capacitor overload. |
| 0x3180 | n503 | Warning: DC Bus link capacitor overload. |
| 0x3210 | F501 | DC Bus link over-voltage. |
| 0x3220 | F502 | DC Bus Link under-voltage. |
| 0x3280 | n502 | Warning: DC Bus Link under-voltage. |
| 0x3281 | n521 | Warning: Dynamic Braking I ² T. |
| 0x3282 | F519 | Regen short circuit. |
| 0x3283 | n501 | Warning: DC Bus link over-voltage. |
| 0x4210 | F234 | Excessive temperature, device (control board). |
| 0x4310 | F235 | Excessive temperature, drive (heat sink). |
| 0x4380 | F236 | Power temperature sensor 2 high. |
| 0x4381 | F237 | Power temperature sensor 3 high. |
| 0x4382 | F535 | Power board overtemperature. |
| 0x4390 | n234 | Warning: Control temperature sensor 1 high. |
| 0x4391 | n235 | Warning: Power temperature sensor 1 high. |
| 0x4392 | n236 | Warning: Power temperature sensor 2 high. |
| 0x4393 | n237 | Warning: Power temperature sensor 3 high. |
| 0x4394 | n240 | Warning: Control temperature sensor 1 low. |
| 0x4395 | n241 | Warning: Power temperature sensor 1 low. |
| 0x4396 | n242 | Warning: Power temperature sensor 2 low. |
| 0x4397 | n243 | Warning: Control temperature sensor 1 low. |
| 0x4398 | F240 | Control temperature sensor 1 low. |
| 0x4399 | F241 | Power temperature sensor 1 low. |
| 0x439A | F242 | Power temperature sensor 2 low. |
| 0x439B | F243 | Power temperature sensor 3 low. |

| Error Code | Fault/ Warning | Description |
|------------|----------------|---|
| 0x5113 | F512 | 5V0 under voltage. |
| 0x5114 | F505 | 1V2 under voltage. |
| 0x5115 | F507 | 2V5 under voltage. |
| 0x5116 | F509 | 3V3 under voltage. |
| 0x5117 | F514 | +12V0 under voltage. |
| 0x5118 | F516 | -12V0 under voltage. |
| 0x5119 | F518 | Analog 3V3 under voltage. |
| 0x5180 | F504 | 1V2 over voltage. |
| 0x5181 | F506 | 2V5 over voltage. |
| 0x5182 | F508 | 3V3 over voltage. |
| 0x5183 | F510 | 5V0 over voltage. |
| 0x5184 | F513 | +12V0 over voltage. |
| 0x5185 | F515 | -12V0 over voltage. |
| 0x5186 | F517 | Analog 3V3 over voltage. |
| 0x5530 | F105 | Hardware memory, non-volatile memory stamp invalid. |
| 0x5580 | F106 | Hardware memory, non-volatile memory data. |
| 0x5590 | F204 | Control board EEPROM read failed. |
| 0x5591 | F205 | Control board EEPROM corrupted serial num stamp. |
| 0x5592 | F206 | Control board EEPROM corrupted serial num data. |
| 0x5593 | F207 | Control board EEPROM corrupted parameter stamp. |
| 0x5594 | F208 | Control board EEPROM corrupted parameter data. |
| 0x5595 | F219 | Control board EEPROM write failed. |
| 0x55A0 | F209 | Power board EEPROM read failed. |
| 0x55A1 | F210 | Power board EEPROM corrupted serial num stamp. |
| 0x55A2 | F212 | Power board EEPROM corrupted serial num data. |
| 0x55A3 | F213 | Power board EEPROM corrupted parameter stamp. |
| 0x55A4 | F214 | Power board EEPROM corrupted parameter data. |
| 0x55A5 | F230 | Power board EEPROM write failed. |
| 0x55A6 | F232 | Power board EEPROM invalid data. |
| 0x55B0 | F248 | Option board EEPROM corrupted. |
| 0x55B1 | F249 | Option board upstream checksum. |
| 0x55B2 | F250 | Option board upstream checksum. |
| 0x55B3 | F251 | Option board watchdog. |
| 0x55B8 | F252 | Firmware and option board FPGA types are not compatible. |
| 0x55B9 | F253 | Firmware and option board FPGA versions are not compatible. |
| 0x55C0 | F621 | Control Board CRC fault. |
| 0x55C1 | F623 | Power Board CRC fault. |
| 0x55C2 | F624 | Power Board Watchdog fault. |

| Error Code | Fault/Warning | Description |
|------------|---------------|---|
| 0x55C3 | F625 | Power Board Communication fault. |
| 0x55C4 | F626 | Power Board FPGA not configured. |
| 0x55C5 | F627 | Control Board Watchdog fault. |
| 0x55C6 | n103 | Warning: Resident FPGA . |
| 0x55C7 | n104 | Warning: Operational FPGA . |
| 0x6380 | F532 | Drive motor parameters setup incomplete. |
| 0x7180 | F301 | Motor overheat. |
| 0x7182 | F305 | Motor Brake open circuit. |
| 0x7183 | F306 | Motor Brake short circuit. |
| 0x7184 | F307 | Motor Brake applied during enable state. |
| 0x7185 | F436 | EnDAT overheated. |
| 0x7186 | n301 | Warning: Motor overheated. |
| 0x7187 | F308 | Voltage exceeds motor rating. |
| 0x7188 | F560 | Regen near capacity, could not prevent over voltage. |
| 0x7189 | F312 | Brake released when it should be applied. |
| 0x7305 | F417 | Broken wire in primary feedback. |
| 0x7380 | F402 | Feedback 1 Analog signal amplitude default. |
| 0x7381 | F403 | Feedback 1 EnDat communication fault. |
| 0x7382 | F404 | Feedback 1 illegal hall state. |
| 0x7383 | F405 | Feedback 1 BiSS watchdog. |
| 0x7384 | F406 | Feedback 1 BiSS multi cycle. |
| 0x7385 | F407 | Feedback 1 BiSS sensor. |
| 0x7386 | F408 | Feedback 1 SFD configuration. |
| 0x7387 | F409 | Feedback 1 SFD UART overrun. |
| 0x7388 | F410 | Feedback 1 SFD UART frame. |
| 0x7389 | F412 | Feedback 1 SFD UART parity. |
| 0x738A | F413 | Feedback 1 SFD transfer timeout. |
| 0x738C | F415 | Feedback 1 SFD mult. corrupt position. |
| 0x738D | F416 | Feedback 1 SFD Transfer incomplete. |
| 0x738E | F418 | Feedback 1 power supply fault. |
| 0x738F | F401 | Feedback 1 failed to set feedback. |
| 0x7390 | n414 | Warning: SFD single corrupted position. |
| 0x7391 | F419 | Encoder init procedure failed. |
| 0x7392 | F534 | Failed to read motor parameters from feedback device. |
| 0x7393 | F421 | SFD position sensor fault. |
| 0x7394 | F463 | Tamagawa encoder: overheat. |
| 0x7395 | n451 | Warning: Tamagawa encoder battery. |
| 0x7396 | n423 | Warning: Non volatile memory error, multiturn overflow. |

| Error Code | Fault/ Warning | Description |
|------------|----------------|--|
| 0x7398 | F135 | Homing is needed. |
| 0x7399 | F468 | FB2.Source not set. |
| 0x739A | F469 | FB1.ENCRES is not power of two. |
| 0x739B | F423 | Non volatile memory error, multiturn overflow. |
| 0x739C | F467 | Hiperface DSL fault. |
| 0x739D | F452 | Multiturn overflow not supported with this feedback. |
| 0x739E | F465 | Excessive shock detected by feedback device. |
| 0x73A0 | F453 | Tamagawa encoder: communication timeout. |
| 0x73A1 | F454 | Tamagawa encoder: communication transfer incomplete. |
| 0x73A2 | F456 | Tamagawa encoder: communication CRC. |
| 0x73A3 | F457 | Tamagawa encoder: communication start timeout. |
| 0x73A4 | F458 | Tamagawa encoder: communication UART overrun. |
| 0x73A5 | F459 | Tamagawa encoder: communication UART framing. |
| 0x73A6 | F460 | Tamagawa encoder: over speed. |
| 0x73A7 | F461 | Tamagawa encoder: contouring error. |
| 0x73A8 | F462 | Tamagawa encoder: counting overflow. |
| 0x73A9 | F464 | Tamagawa encoder: multiturn error. |
| 0x73AA | F451 | Tamagawa encoder: battery. |
| 0x73B0 | F486 | Motor velocity exceeds emulated encoder maximum speed. |
| 0x73B8 | F420 | FB3 EnDat communication fault. |
| 0x73C5 | F482 | Commutation not initialized. |
| 0x73C6 | F483 | Motor U phase missing. |
| 0x73C7 | F484 | Motor V phase missing. |
| 0x73C8 | F485 | Motor W phase missing. |
| 0x73D0 | F487 | Wake and Shake. Validating positive movement failed. |
| 0x73D1 | F489 | Wake and Shake. Validating negative movement failed. |
| 0x73D2 | F490 | Wake and Shake. Validating commutation angle time out. |
| 0x73D3 | F491 | Wake and Shake. Validating commutation angle moved too far. |
| 0x73D4 | F492 | Wake and Shake. Validating commutation angle required more than MOTOR.ICONT. |
| 0x73D5 | F493 | Invalid commutation detected, motor accelerates in wrong direction. |
| 0x8130 | F129 | Life Guard Error or Heartbeat Error. |
| 0x8180 | n702 | Warning: Fieldbus communication lost. |
| 0x8280 | n601 | Warning: Modbus data rate is too high. |
| 0x8311 | F304 | Motor foldback. |
| 0x8331 | F524 | Drive foldback. |
| 0x8380 | n524 | Warning: Drive foldback. |

| Error Code | Fault/Warning | Description |
|------------|---------------|---|
| 0x8381 | n304 | Warning: Motor foldback. |
| 0x8382 | n309 | Warning: Motor I ² t load. |
| 0x8383 | n580 | Warning: Using derivate of position when using sensorless feedback type in position mode. |
| 0x8384 | n581 | Warning: Zero velocity when using induction sensorless feedback type in position mode. |
| 0x8480 | F302 | Over speed. |
| 0x8481 | F703 | Emergency timeout occurred while axis should disable. |
| 0x8482 | F480 | Fieldbus command velocity too high. |
| 0x8483 | F481 | Fieldbus command velocity too low. |
| 0x8582 | n107 | Warning: Positive software position limit is exceeded. |
| 0x8583 | n108 | Warning: Negative software position limit is exceeded. |
| 0x8611 | F439 | Following error (user). |
| 0x8685 | F138 | Instability during autotune. |
| 0x8686 | n151 | Warning: Not enough distance to move; Motion Exception. |
| 0x8687 | n152 | Warning: Not enough distance to move; Following Motion Exception. |
| 0x8688 | n153 | Warning: Velocity Limit Violation, Exceeding Max Limit. |
| 0x8689 | n154 | Warning: Following Motion Failed; Check Motion Parameters. |
| 0x868A | n156 | Warning: Target Position crossed due to Stop command. |
| 0x86A0 | n157 | Warning: Homing Index pulse not found. |
| 0x86A1 | n158 | Warning: Homing Reference Switch not found. |
| 0x86A2 | n159 | Warning: Failed to set motion task parameters. |
| 0x86A3 | n160 | Warning: Motion Task Activation Failed. |
| 0x86A4 | n161 | Warning: Homing Procedure Failed. |
| 0x86A5 | F139 | Target Position Over Short due to invalid Motion task activation. |
| 0x86A6 | n163 | Warning: MT.NUM exceeds limit. |
| 0x86A7 | n164 | Warning: Motion task is not initialized. |
| 0x86A8 | n165 | Warning: Motion task target position is out. |
| 0x86A9 | n167 | Warning: Software limit switch traversed. |
| 0x86AA | n168 | Warning: Invalid bit combination in the motion task control word. |
| 0x86AB | n169 | Warning: 1:1 profile cannot be triggered on the fly. |
| 0x86AC | n170 | Warning: Customer profile table is not initialized. |
| 0x86AD | n171 | Warning: Motion task activation is currently pending |
| 0x86AE | n135 | Warning: Homing is needed. |
| 0x86AF | n174 | Warning: Homing maximum distance exceeded |
| 0x86B0 | F438 | Following error (numeric). |
| 0x8780 | F125 | Fieldbus synchronization lost. |
| 0x8781 | n125 | Warning: Fieldbus synchronization lost. |

| Error Code | Fault/Warning | Description |
|------------|---------------|---|
| 0x8AF0 | n137 | Warning: Homing and feedback mismatch. |
| 0xFF00 | F701 | Fieldbus runtime. |
| 0xFF01 | F702 | Fieldbus communication lost. |
| 0xFF02 | F529 | Iu current offset limit exceeded. |
| 0xFF03 | F530 | Iv current offset limit exceeded. |
| 0xFF04 | F521 | Regen over power. |
| 0xFF07 | F525 | Output over current. |
| 0xFF08 | F526 | Current sensor short circuit. |
| 0xFF09 | F128 | MPOLES/FPOLES not an integer. |
| 0xFF0A | F531 | Power stage fault. |
| 0xFF0B | F602 | Safe torque off. |
| 0xFF0C | F131 | Secondary feedback A/B line break. |
| 0xFF0D | F130 | Secondary feedback supply over current. |
| 0xFF0E | F134 | Secondary feedback illegal state. |
| 0xFF0F | F245 | External fault. |
| 0xFF10 | F136 | Firmware and FPGA versions are not compatible. |
| 0xFF11 | F101 | Firmware type mismatch. |
| 0xFF12 | n439 | Warning: Following error (user). |
| 0xFF13 | n438 | Warning: Following error (numeric). |
| 0xFF14 | n102 | Warning: Operational FPGA is not a default FPGA. |
| 0xFF15 | n101 | Warning: The FPGA is a laboratory FPGA. |
| 0xFF16 | n602 | Warning: Safe torque off. |
| 0xFF17 | F132 | Secondary feedback Z line break. |
| 0xFF18 | n603 | Warning: OPMODE incompatible with CMDSOURCE. |
| 0xFF19 | n604 | Warning: EMUEMODE incompatible with DRV.HANDWHEELSRC. |

19.33.2 Object Dictionary

The following tables describe all objects reachable via SDO or PDO. (i.p. = in preparation).

Abbreviations:

| | | | |
|--------|------------------|-------|------------------|
| U | = UNSIGNED | RO | = Read only |
| INT | = INTEGER | RW | = Read and Write |
| VisStr | = Visible String | WO | = Write only |
| | | const | = Constant |

19.33.2.1 Float Scaling

The scaling applied to objects which match floating-point parameters in EWV¹/Telnet are listed in the column "Float Scaling."

For example, index 607Ah is listed as 1:1 - this means that commanding a value of 1000 in SDO 607Ah is equivalent to entering MT.P 1000.000 in EWV. On the other hand, index 3598h is listed as 1000:1 - this means that commanding a value of 1000 in SDO 3598h is equivalent to entering IL.KP 1.000 in EWV.

A few parameters are listed as variable (var), because the scaling depends on other settings.

19.33.2.2 Effectiveness of PDO set-points

Some objects are having effect only in the state machine state "Operation Enabled", which is controlled by the DS402 control word. They are marked with an asterisk (*) at the PDO mapping.

19.33.2.3 Communication SDOs

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|--------------|
| 1000h | 0 | U32 | | RO | no | Device type | — |
| 1001h | 0 | U8 | | RO | no | Error register | — |
| 1002h | 0 | U32 | | RO | yes | Manufacturer-specific status register | — |
| 1003h | | ARRAY | | | | Pre-defined error field | — |
| 1003h | 0 | U8 | | RW | no | Number of errors | — |
| 1003h | 1 to 10 | U32 | | RO | no | standard error field | — |
| 1005h | 0 | U32 | | RW | no | COB—ID SYNC message | — |
| 1006h | 0 | U32 | | RW | no | Communication cycle period | — |
| 1008h | 0 | VisStr | | const | no | Manufacturer device name | — |
| 1009h | 0 | VisStr | | const | no | Manufacturer hardware version | — |
| 100Ah | 0 | VisStr | | const | no | Manufacturer software version | — |
| 100Ch | 0 | U16 | | RW | no | Guard time | — |
| 100Dh | 0 | U8 | | RW | no | Lifetime factor | — |
| 1010h | | ARRAY | | | | Save parameters | — |
| 1010h | 0 | U8 | | RO | no | highest sub-index | — |
| 1010h | 1 | U32 | | RW | no | Saves the drive parameters from the RAM to the NV. | DRV.NVSAVE |
| 1011h | | ARRAY | | | | Load parameters | — |
| 1011h | 0 | U8 | | RO | no | highest sub-index | — |
| 1011h | 1 | U32 | | RW | no | Loads default parameters to the RAM. | DRV.RSTVAR |

¹Embedded Workbench Views

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|-------------------------------------|--------------|
| 1012h | 0 | U32 | | RW | no | COB—ID for the Time Stamp | — |
| 1014h | 0 | U32 | | RW | no | COB—ID for the Emergency Object | — |
| 1016h | | RECORD | | | | Consumer heartbeat time | |
| 1016h | 0 | U8 | | RO | no | highest sub-index | — |
| 1016h | 1 | U32 | | RW | no | Consumer heartbeat time | — |
| 1017h | 0 | U16 | | RW | no | Producer heartbeat time | — |
| 1018h | | RECORD | | | | Identity Object | — |
| 1018h | 0 | U8 | | RO | no | highest sub-index | — |
| 1018h | 1 | U32 | | RO | no | Vendor ID | — |
| 1018h | 2 | U32 | | RO | no | Product Code | — |
| 1018h | 3 | U32 | | RO | no | Revision number | — |
| 1018h | 4 | U32 | | RO | no | Serial number | — |
| 1026h | | ARRAY | | | | OS prompt | — |
| 1026h | 0 | U8 | | RO | no | highest sub-index | — |
| 1026h | 1 | U8 | | WO | no | StdIn | — |
| 1026h | 2 | U8 | | RO | no | StdOut | — |
| 1400h | | RECORD | | | | RXPDO1 communication parameter | — |
| 1400h | 0 | U8 | | RO | no | highest sub-index | — |
| 1400h | 1 | U32 | | RW | no | RXPDO1 COB — ID | — |
| 1400h | 2 | U8 | | RW | no | Transmission type RXPDO1 | — |
| 1401h | | RECORD | | | | RXPDO2 communication parameter | — |
| 1401h | 0 | U8 | | RO | no | highest sub-index | — |
| 1401h | 1 | U32 | | RW | no | RXPDO2 COB—ID | — |
| 1401h | 2 | U8 | | RW | no | Transmission type RXPDO2 | — |
| 1402h | | RECORD | | | | RXPDO3 communication parameter | — |
| 1402h | 0 | U8 | | RO | no | highest sub-index | — |
| 1402h | 1 | U32 | | RW | no | RXPDO3 COB—ID | — |
| 1402h | 2 | U8 | | RW | no | Transmission type RXPDO3 | — |
| 1403h | | RECORD | | | | RXPDO4 communication parameter | — |
| 1403h | 0 | U8 | | RO | no | highest sub-index | — |
| 1403h | 1 | U32 | | RW | no | RXPDO4 COB—ID | — |
| 1403h | 2 | U8 | | RW | no | Transmission type RXPDO4 | — |
| 1600h | | RECORD | | | | RXPDO1 mapping parameter | — |
| 1600h | 0 | U8 | | RO | no | highest sub-index | — |
| 1600h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1601h | | RECORD | | | | RXPDO2 mapping parameter | — |
| 1601h | 0 | U8 | | RO | no | highest sub-index | — |
| 1601h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1602h | | RECORD | | | | RXPDO3 mapping parameter | — |
| 1602h | 0 | U8 | | RO | no | highest sub-index | — |
| 1602h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|-------------------------------------|--------------|
| 1603h | | RECORD | | | | RXPDO4 mapping parameter | — |
| 1603h | 0 | U8 | | RO | no | highest sub-index | — |
| 1603h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1800h | | RECORD | | | | TXPDO1 communication parameter | — |
| 1800h | 0 | U8 | | RO | no | highest sub-index | — |
| 1800h | 1 | U32 | | RW | no | TXPDO1 COB—ID | — |
| 1800h | 2 | U8 | | RW | no | Transmission type TXPDO1 | — |
| 1800h | 3 | U16 | | RW | no | Inhibit time | — |
| 1800h | 4 | U8 | | const | no | reserved | — |
| 1800h | 5 | U16 | | RW | no | Event timer | — |
| 1801h | | RECORD | | | | TXPDO2 communication parameter | — |
| 1801h | 0 | U8 | | RO | no | highest sub-index | — |
| 1801h | 1 | U32 | | RW | no | TXPDO2 COB—ID | — |
| 1801h | 2 | U8 | | RW | no | Transmission type TXPDO2 | — |
| 1801h | 3 | U16 | | RW | no | Inhibit time | — |
| 1801h | 4 | U8 | | const | no | reserved | — |
| 1801h | 5 | U16 | | RW | no | Event timer | — |
| 1802h | | RECORD | | | | TXPDO3 communication parameter | — |
| 1802h | 0 | U8 | | RO | no | highest sub-index | — |
| 1802h | 1 | U32 | | RW | no | TXPDO3 COB—ID | — |
| 1802h | 2 | U8 | | RW | no | Transmission type TXPDO3 | — |
| 1802h | 3 | U16 | | RW | no | Inhibit time | — |
| 1802h | 4 | U8 | | const | no | reserved | — |
| 1802h | 5 | U16 | | RW | no | Event timer | — |
| 1803h | | RECORD | | | | TXPDO4 communication parameter | — |
| 1803h | 0 | U8 | | RO | no | highest sub-index | — |
| 1803h | 1 | U32 | | RW | no | TXPDO4 COB—ID | — |
| 1803h | 2 | U8 | | RW | no | Transmission type TXPDO4 | — |
| 1803h | 3 | U16 | | RW | no | Inhibit time | — |
| 1803h | 4 | U8 | | const | no | reserved | — |
| 1803h | 5 | U16 | | RW | no | Event timer | — |
| 1A00h | | RECORD | | | | Mapping parameter TXPDO1 | — |
| 1A00h | 0 | U8 | | RO | no | highest sub-index | — |
| 1A00h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1A01h | | RECORD | | | | Mapping parameter TXPDO2 | — |
| 1A01h | 0 | U8 | | RO | no | highest sub-index | — |
| 1A01h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1A02h | | RECORD | | | | Mapping parameter TXPDO3 | — |
| 1A02h | 0 | U8 | | RO | no | highest sub-index | — |
| 1A02h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1A03h | | RECORD | | | | Mapping parameter TXPDO4 | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--------------------------------------|--------------|
| 1A03h | 0 | U8 | | RO | no | highest sub-index | — |
| 1A03h | 1 to 8 | U32 | | RW | no | Mapping for n—the application object | — |
| 1C12h | | ARRAY | | RW | no | RxPDO assign | — |
| 1C12h | 0 | U8 | | RO | no | highest sub-index | — |
| 1C13h | 1 to 4 | U8 | | RW | no | Subindex 001..004 | — |
| 1C13h | | ARRAY | | RW | no | TxPDO assign | — |
| 1C13h | 0 | U8 | | RO | no | highest sub-index | — |
| 1C13h | 1 to 4 | U8 | | RW | no | Subindex 001..004 | — |

19.33.2.4 Manufacturer specific SDOs

Objects 2000h to 3999h

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|---------------|
| 2000h | | ARRAY | | | | System Warnings | — |
| 2000h | 0 | U8 | | RO | no | highest sub-index | — |
| 2000h | 1 | U32 | | RO | no | System Warning 1 | DRV.WARNING1 |
| 2000h | 2 | U32 | | RO | no | System Warning 2 | DRV.WARNING2 |
| 2000h | 3 | U32 | | RO | no | System Warning 3 | DRV.WARNING3 |
| 2001h | | ARRAY | | | | System Faults | — |
| 2001h | 0 | U8 | | RO | no | highest sub-index | — |
| 2001h | 1 | U32 | | RO | no | System Fault 1 | DRV.FAULT1 |
| 2001h | 2 | U32 | | RO | no | System Fault 2 | DRV.FAULT2 |
| 2001h | 3 | U32 | | RO | no | System Fault 3 | DRV.FAULT3 |
| 2001h | 4 | U32 | | RO | no | System Fault 4 | DRV.FAULT4 |
| 2001h | 5 | U32 | | RO | no | System Fault 5 | DRV.FAULT5 |
| 2001h | 6 | U32 | | RO | no | System Fault 6 | DRV.FAULT6 |
| 2001h | 7 | U32 | | RO | no | System Fault 7 | DRV.FAULT7 |
| 2001h | 8 | U32 | | RO | no | System Fault 8 | DRV.FAULT8 |
| 2001h | 9 | U32 | | RO | no | System Fault 9 | DRV.FAULT9 |
| 2001h | A | U32 | | RO | no | System Fault 10 | DRV.FAULT10 |
| 2002h | | ARRAY | | | | Manufacturer status bytes | — |
| 2002h | 0 | U8 | | RO | no | highest sub-index | — |
| 2002h | 1 | U8 | | RO | yes | Manufacturer status bytes 1 | — |
| 2002h | 2 | U8 | | RO | yes | Manufacturer status bytes 2 | — |
| 2002h | 3 | U8 | | RO | yes | Manufacturer status bytes 3 | — |
| 2002h | 4 | U8 | | RO | yes | Manufacturer status bytes 4 | — |
| 2011h | | VAR | | RO | | DRV.RUNTIME in seconds | DRV.RUNTIME |
| 2012h | | ARRAY | | | | Fault history: Fault numbers | DRV.FAULTHIST |
| 2012h | 0 | U8 | | RO | no | highest sub-index | — |
| 2012h | 1 to 20 | U32 | | RO | no | Nth-latest entry in fault number list of fault history table | — |
| 2013h | | ARRAY | | | | Fault history: Time stamps | DRV.FAULTHIST |
| 2013h | 0 | U8 | | RO | no | highest sub-index | — |
| 2013h | 1 to 20 | U32 | | RO | no | Nth-latest entry in fault time stamp list of fault history table | — |
| 2014h | | ARRAY | | | | Mask TxPDO Channel 1 | — |
| 2014h | 1 | U32 | | RW | no | Mask (Byte 0..3) | — |
| 2014h | 2 | U32 | | RW | no | Mask (Byte 4..7) | — |
| 2015h | | ARRAY | | | | Mask TxPDO Channel 2 | — |
| 2015h | 1 | U32 | | RW | no | Mask (Byte 0..3) | — |
| 2015h | 2 | U32 | | RW | no | Mask (Byte 4..7) | — |
| 2016h | | ARRAY | | | | Mask TxPDO Channel 3 | — |
| 2016h | 1 | U32 | | RW | no | Mask (Byte 0..3) | — |
| 2016h | 2 | U32 | | RW | no | Mask (Byte 4..7) | — |
| 2017h | | ARRAY | | | | Mask TxPDO Channel 4 | — |
| 2017h | 1 | U32 | | RW | no | Mask (Byte 0..3) | — |
| 2017h | 2 | U32 | | RW | no | Mask (Byte 4..7) | — |
| 2018h | | ARRAY | | | | Firmware version | — |
| 2018h | 0 | U16 | | const | no | highest sub-index | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|-----------------------|
| 2018h | 1 | U16 | | const | no | Major version | — |
| 2018h | 2 | U16 | | const | no | Minor version | — |
| 2018h | 3 | U16 | | const | no | Revision | — |
| 2018h | 4 | U16 | | const | no | Branch version | — |
| 2026h | | ARRAY | | | | ASCII Channel | — |
| 2026h | 0 | U8 | | RO | no | highest sub-index | — |
| 2026h | 1 | VisStr | | WO | no | Command | — |
| 2026h | 2 | VisStr | | RO | no | Response | — |
| 2031h | 0 | VisStr | | RW | no | Drive Name, length 10 bits | DRV.NAME |
| 2032h | 0 | VisStr | | RW | no | Drive custom identifier string, length 32 byte | DRV.CUSTOM-IDENTIFIER |
| 204Ch | | ARRAY | | | | pv scaling factor | — |
| 204Ch | 0 | U8 | | RO | no | highest sub-index | — |
| 204Ch | 1 | INT32 | | RW | no | pv scaling factor numerator | — |
| 204Ch | 2 | INT32 | | RW | no | pv scaling factor denominator | — |
| 2050h | 0 | INT32 | 1:1 | RO | yes | Position, secondary feedback | DRV.HANDWHEEL |
| 2071h | 0 | INT32 | | RW | yes* | Target current | - |
| 2077h | 0 | INT32 | | RO | yes | Current actual value | - |
| 20A0h | 0 | INT32 | var | RO | yes | Latch position 1, positive edge | CAP0.PLFB , CAP0.T |
| 20A1h | 0 | INT32 | var | RO | yes | Latch position 1, negative edge | CAP0.PLFB , CAP0.T |
| 20A2h | 0 | INT32 | var | RO | yes | Latch position 2, positive edge | CAP1.PLFB , CAP1.T |
| 20A3h | 0 | INT32 | var | RO | yes | Latch position 2, negative edge | CAP1.PLFB , CAP1.T |
| 20A4h | 0 | U16 | | RW | yes | Latch control register | — |
| 20A5h | 0 | U16 | | RW | yes | Latch status register | — |
| 20A6h | 0 | INT32 | var | RO | yes | Gets captured position value | CAP0.PLFB |
| 20A7h | 0 | INT32 | var | RO | yes | Gets captured position value | CAP1.PLFB |
| 20B8h | 0 | U16 | | RW | yes | Clear changed digital input information | — |
| 3405h | | ARRAY | | | | VL.ARTYPE | — |
| 3405h | 0 | U8 | | RO | no | highest sub-index | — |
| 3405h | 1 | U8 | | RW | no | Calculation method for BiQuad filter 1 | VL.ARTYPE1 |
| 3405h | 2 | U8 | | RW | no | Calculation method for BiQuad filter 2 | VL.ARTYPE2 |
| 3405h | 3 | U8 | | RW | no | Calculation method for BiQuad filter 3 | VL.ARTYPE3 |
| 3405h | 4 | U8 | | RW | no | Calculation method for BiQuad filter 4 | VL.ARTYPE4 |
| 3406h | | ARRAY | | | | VL BiQuad | — |
| 3406h | 0 | U8 | | RO | no | highest sub-index | — |
| 3406h | 1 | U32 | 1000:1 | RW | no | Natural frequency of pole of anti-resonance (AR) filter 1 | VL.ARPF1 |
| 3406h | 2 | U32 | 1000:1 | RW | no | Natural frequency of pole of anti-resonance (AR) filter 2 | VL.ARPF2 |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|----------------------|
| 3406h | 3 | U32 | 1000:1 | RW | no | Natural frequency of pole of anti-resonance (AR) filter 3 | VL.ARPFF3 |
| 3406h | 4 | U32 | 1000:1 | RW | no | Natural frequency of pole of anti-resonance (AR) filter 4 | VL.ARPFF4 |
| 3406h | 5 | U32 | 1000:1 | RW | no | Q of pole of anti-resonance (AR) filter 1 | VL.ARPQ1 |
| 3406h | 6 | U32 | 1000:1 | RW | no | Q of pole of anti-resonance (AR) filter 2 | VL.ARPQ2 |
| 3406h | 7 | U32 | 1000:1 | RW | no | Q of pole of anti-resonance (AR) filter 3 | VL.ARPQ3 |
| 3406h | 8 | U32 | 1000:1 | RW | no | Q of pole of anti-resonance (AR) filter 4 | VL.ARPQ4 |
| 3406h | 9 | U32 | 1000:1 | RW | no | Natural frequency of zero of anti-resonance (AR) filter 1 | VL.ARZF1 |
| 3406h | A | U32 | 1000:1 | RW | no | Natural frequency of zero of anti-resonance (AR) filter 2 | VL.ARZF2 |
| 3406h | B | U32 | 1000:1 | RW | no | Natural frequency of zero of anti-resonance (AR) filter 3 | VL.ARZF3 |
| 3406h | C | U32 | 1000:1 | RW | no | Natural frequency of zero of anti-resonance (AR) filter 4 | VL.ARZF4 |
| 3406h | D | U32 | 1000:1 | RW | no | Q of zero of anti-resonance filter 1 | VL.ARZQ1 |
| 3406h | E | U32 | 1000:1 | RW | no | Q of zero of anti-resonance filter 2 | VL.ARZQ2 |
| 3406h | F | U32 | 1000:1 | RW | no | Q of zero of anti-resonance filter 3 | VL.ARZQ3 |
| 3406h | 10 | U32 | 1000:1 | RW | no | Q of zero of anti-resonance filter 4 | VL.ARZQ4 |
| 3407h | | STRUCT | | | | Velocity Filter | — |
| 3407h | 0 | U8 | | RO | no | highest sub-index | — |
| 3407h | 1 | INT32 | 1000:1 | RW | no | 10 Hz filtered VL.FB | VL.FBFILTER |
| 3407h | 2 | U32 | 1000:1 | RW | no | Gain for the velocity feed-forward | VL.KVFF |
| 3407h | 3 | U32 | | RW | no | Gain for the acceleration feed-forward | VL.KBUSFF |
| 3407h | 4 | U32 | 1:1 | RW | no | Sets the velocity error | VL.ERR |
| 3412h | 0 | INT8 | | RW | no | Type of regen resistor | REGEN.TYPE |
| 3414h | 0 | U8 | | RW | | Returns and sets the regen resistor fault level temperature. | REGEN.WATTEXT |
| 3415h | 0 | U32 | 1000:1 | RO | no | Thermal regen resistor time constant | REGEN.TEXT |
| 3416h | 0 | U32 | | RO | no | Gets regen resistor's calculated power | REGEN.POWER |
| 3417h | 0 | U32 | | RO | no | Returns a filtered version of 3416h | REGEN.POWER-FILTERED |
| 3420h | 0 | U16 | 1000:1 | RW | no | Sets the foldback fault level. | IL.FOLDFTHRESH |
| 3421h | 0 | U32 | 1000:1 | RW | no | Sets the user value for the foldback fault level. | IL.FOLDFTHRESHU |
| 3422h | 0 | U32 | 1000:1 | | no | Sets friction compensation value. | IL.FRICTION |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|-------------------------------|
| 3423h | 0 | INT32 | 1000:1 | | no | A constant current command added to compensate for gravity. | IL.OFFSET |
| 3424h | 0 | U16 | | | no | Enables/disables the integrator part of the PI loop. | IL.INTEN (Password Protected) |
| 3425h | 0 | U32 | 1000:1 | RO | no | Reads the overall foldback current limit | IL.IFOLD |
| 3426h | 0 | U32 | 1000:1 | RW | no | Sets current loop acceleration feedforward gain value | IL.KACFF |
| 3427h | | RECORD | | | | Motor protection parameters | — |
| 3427h | 0 | U8 | | RO | no | highest sub-index | — |
| 3427h | 1 | U8 | | RW | no | | IL.MIMODE |
| 3427h | 2 | U8 | | RW | no | | IL.MI2TWTHRESH |
| 3427h | 3 | U32 | | RW | yes | | IL.MI2T |
| 3430h | 0 | U8 | | RW | no | Sets the direction for absolute motion tasks. | PL.MODPDIR |
| 3431h | 0 | U16 | | RW | no | Sets the motion task in the drive | MT.SET |
| 3440h | | ARRAY | | | | Controlled stop parameters | — |
| 3440h | 0 | U8 | | RO | no | highest sub-index | — |
| 3440h | 1 | U32 | 1:1 | RW | no | Sets the deceleration value for a controlled stop. | CS.DEC |
| 3440h | 2 | U32 | 1:1 | RW | no | Sets the velocity threshold for a controlled stop. | CS.VTHRESH |
| 3440h | 3 | U32 | | RW | no | Sets the time value for the drive velocity to be within CS.VTHRESH. | CS.TO |
| 3441h | 0 | U8 | | RO | no | Controlled stop state | CS.STATE |
| 3443h | 0 | U16 | | RO | no | Returns the possible reason for a drive disable | DRV.DIS |
| 3444h | 0 | U16 | 1000:1 | RO | no | Maximum current for dynamic braking | DRV.DBILIMIT |
| 3445h | 0 | U32 | | RO | no | Emergency timeout for braking | DRV.DISTO |
| 3450h | 0 | U8 | | WO | no | Release or enable brake | MOTOR.BRAKERLS |
| 3451h | 0 | U8 | | RW | no | Determines which drive parameters are calculated automatically. | MOTOR.AUTASET |
| 3452h | 0 | U16 | | RW | no | Sets the motor maximum voltage | MOTOR.VOLTMAX |
| 3453h | 0 | U32 | | RW | no | Sets the motor temperature warning level | MOTOR.TEMPWARN |
| 3454h | 0 | U32 | 1000:1 | RW | no | Sets the thermal constant of the motor coil | MOTOR.CTF0 |
| 3455h | 0 | U32 | 1000:1 | RW | no | Sets the line-to-line motor Lq | MOTOR.LQLL |
| 3456h | 0 | U32 | 1000:1 | RW | no | Sets the stator winding resistance phase-phase in ohms | MOTOR.R |
| 3457h | | RECORD | | | | Induction Motor parameter | — |
| 3457h | 0 | U8 | | RO | no | highest sub-index | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|-----------------|
| 3457h | 1 | INT32 | 1000:1 | RW | no | Configuration of induction motor's rated velocity. | MOTOR.VRATED |
| 3457h | 2 | U16 | | RW | no | Configuration of induction motor's rated voltage. | MOTOR.VOLTRATED |
| 3457h | 3 | U16 | | RW | no | Sets the minimum voltage for V/f Control. | MOTOR.VOLTMIN |
| 345Ah | | ARRAY | | | | Brake Control | — |
| 345Ah | 0 | U8 | | RO | no | highest sub-index | — |
| 345Ah | 1 | U16 | | RW | yes | Brake Control Command | — |
| 345Ah | 2 | U16 | | RO | yes | Brake Status Response. | — |
| 3460h | | RECORD | | | | Capture engines parameters | — |
| 3460h | 0 | U8 | | RO | no | highest sub-index | — |
| 3460h | 1 | U8 | | RW | no | Specifies the trigger source for the position capture. | CAP0.TRIGGER |
| 3460h | 2 | U8 | | RW | no | Specifies the trigger source for the position capture. | CAP1.TRIGGER |
| 3460h | 3 | U8 | | RW | no | Selects the captured value. | CAP0.MODE |
| 3460h | 4 | U8 | | RW | no | Selects the captured value. | CAP1.MODE |
| 3460h | 5 | U8 | | RW | no | Controls the precondition logic. | CAP0.EVENT |
| 3460h | 6 | U8 | | RW | no | Controls the precondition logic. | CAP1.EVENT |
| 3460h | 7 | U8 | | RW | no | Selects the capture precondition edge. | CAP0.PREEDGE |
| 3460h | 8 | U8 | | RW | no | Selects the capture precondition edge. | CAP1.PREEDGE |
| 3460h | 9 | U8 | | RW | no | Sets the precondition trigger. | CAP0.PRESELECT |
| 3460h | A | U8 | | RW | no | Sets the precondition trigger. | CAP1.PRESELECT |
| 3460h | B | U8 | | RW | no | Selects the feedback source for the capture engine 0. | CAP0.FBSOURCE |
| 3460h | C | U8 | | RW | no | Selects the feedback source for the capture engine 1. | CAP1.FBSOURCE |
| 3470h | | RECORD | | | | | — |
| 3470h | 0 | U8 | | RO | no | highest sub-index | — |
| 3470h | 1 | INT8 | | RW | no | Sets the analog output mode. | AOUT.MODE |
| 3470h | 2 | INT16 | 1000:1 | RW | yes | Reads the analog output value. | AOUT.VALUE |
| 3470h | 3 | INT16 | 1000:1 | RW | yes | Reads and writes the analog output value. | AOUT.VALUEU |
| 3470h | 4 | INT16 | 1000:1 | RO | yes | Reads the value of the analog input signal. | AIN.VALUE |
| 3470h | 5 | U32 | 1000:1 | RW | no | Sets velocity scale factor for analog output | AOUT.VSCALE |
| 3471h | 0 | U32 | 1:1 | RW | no | Sets the analog position scale factor | AOUT.PSCALE |
| 3472h | 0 | U32 | 1:1 | RW | no | Sets analog pscale factor | AIN.PSCALE |
| 3474h | | ARRAY | | | | DINx.PARAM | — |
| 3474h | 0 | U8 | | RO | no | highest sub-index | — |
| 3474h | 1 | U32 | | RW | no | Lower 32-bit part of input parameter 1 | DIN1.PARAM |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|--------------|
| 3474h | 2 | U32 | | RW | no | Lower 32-bit part of input parameter 2 | DIN2.PARAM |
| 3474h | 3 | U32 | | RW | no | Lower 32-bit part of input parameter 3 | DIN3.PARAM |
| 3474h | 4 | U32 | | RW | no | Lower 32-bit part of input parameter 4 | DIN4.PARAM |
| 3474h | 5 | U32 | | RW | no | Lower 32-bit part of input parameter 5 | DIN5.PARAM |
| 3474h | 6 | U32 | | RW | no | Lower 32-bit part of input parameter 6 | DIN6.PARAM |
| 3474h | 7 | U32 | | RW | no | Lower 32-bit part of input parameter 7 | DIN7.PARAM |
| 3474h | 8 | U32 | | RW | no | Higher 32-bit part of input parameter 1 | DIN1.PARAM |
| 3474h | 9 | U32 | | RW | no | Higher 32-bit part of input parameter 2 | DIN2.PARAM |
| 3474h | A | U32 | | RW | no | Higher 32-bit part of input parameter 3 | DIN3.PARAM |
| 3474h | 4 | U32 | | RW | no | Higher 32-bit part of input parameter 1 | DIN1.PARAM |
| 3474h | 5 | U32 | | RW | no | Higher 32-bit part of input parameter 2 | DIN2.PARAM |
| 3474h | 6 | U32 | | RW | no | Higher 32-bit part of input parameter 3 | DIN3.PARAM |
| 3474h | B | U32 | | RW | no | Higher 32-bit part of input parameter 4 | DIN4.PARAM |
| 3474h | C | U32 | | RW | no | Higher 32-bit part of input parameter 5 | DIN5.PARAM |
| 3474h | D | U32 | | RW | no | Higher 32-bit part of input parameter 6 | DIN6.PARAM |
| 3474h | E | U32 | | RW | no | Higher 32-bit part of input parameter 7 | DIN7.PARAM |
| 3475h | | ARRAY | | | | DOUTx.PARAM | — |
| 3475h | 0 | U8 | | RO | no | highest sub-index | — |
| 3475h | 1 | U32 | | RW | no | Lower 32-bit part of output parameter 1 | DOUT1.PARAM |
| 3475h | 2 | U32 | | RW | no | Lower 32-bit part of output parameter 2 | DOUT2.PARAM |
| 3475h | 3 | U32 | | RW | no | Higher 32-bit part of output parameter 1 | DOUT1.PARAM |
| 3475h | 2 | U32 | | RW | no | Higher 32-bit part of output parameter 1 | DOUT1.PARAM |
| 3475h | 4 | U32 | | RW | no | Higher 32-bit part of output parameter 2 | DOUT2.PARAM |
| 3480h | 0 | U32 | 1000:1 | RW | no | Integral gain of position regulator PID loop | PL.KI |
| 3481h | | ARRAY | | | | PL.INTMAX | — |
| 3481h | 0 | U8 | | RO | no | highest sub-index | — |
| 3481h | 1 | U32 | 1:1 | RW | no | Input saturation | PL.INTINMAX |
| 3481h | 2 | U32 | 1:1 | RW | no | Output saturation | PL.INTOUTMAX |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|-----------------|
| 3482h | 0 | INT32 | 1:1 | RO | no | Maximum value of following error in homing | HOME.PERRTHRESH |
| 3483h | 0 | INT32 | 1:1 | RW | no | Sets the position error warning level | PL.ERRWTHRESH |
| 3484h | 0 | INT32 | 1:1 | RW | no | Specification of an additional movement after homing is completed. | HOME.DIST |
| 3490h | 0 | INT32 | 1:1 | RW | no | Position feedback offset | FB1.OFFSET |
| 3491h | 0 | U32 | | RO | no | Location of index pulse on EEO | DRV.EMUEMTURN |
| 3492h | 0 | U32 | | RO | no | Motion status of the drive | DRV.MOTIONSTAT |
| 3493h | 0 | U8 | | RO | no | Direction of EEO (emulated encoder output) | DRV.EMUEDIR |
| 3494h | | RECORD | | | | WS parameters | — |
| 3494h | 0 | U8 | | RO | no | highest sub-index | — |
| 3494h | 1 | INT16 | 1000:1 | RW | no | Sets maximum current used for wake and shake | WS.IMAX |
| 3494h | 2 | INT32 | 1:1 | RW | no | Sets the maximum movement required for wake and shake | WS.DISTMAX |
| 3494h | 3 | U16 | | RW | no | Sets the delay for wake and shake between loops in mode 0 | WS.TDELAY30 |
| 3494h | 4 | INT32 | 1:1 | RW | no | Defines the maximum allowed velocity for Wake & Shake | WS.VTHRESH |
| 3494h | 5 | U8 | | RO | no | Reads wake and shake status | WS.STATE |
| 3494h | 6 | U8 | | RW | no | Arm Wake and Shake to start | WS.ARM |
| 3495h | 0 | U16 | 1000:1 | RW | no | Voltage level for undervoltage warning. | VBUS.UVWTHRESH |
| 3496h | | ARRAY | | | | FBUS synchronization parameters | — |
| 3496h | 0 | U8 | | RO | no | highest sub-index | — |
| 3496h | 1 | U32 | | RW | no | expected time distance between clearing the PLL counter and calling the PLL function | FBUS.SYNCDIST |
| 3496h | 2 | U32 | | RW | no | actual time distance between clearing the PLL counter and calling the PLL function | FBUS.SYNCACT |
| 3496h | 3 | U32 | | RW | no | Time window, which is used in order to consider the drive as being synchronized | FBUS.SYNCWND |
| 3496h | 4 | U32 | | RW | no | Time, which is used for extending or lowering the sample rate of the internal 16 [kHz] IRQ | — |
| 3498h | 0 | U8 | | RW | no | Protection level of fieldbus against other communication channels (Telnet, Modbus..) | FBUS.PROTECTION |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|-----------------|
| 3499h | 0 | INT32 | | RW | yes | Set-point for stepper motor output through the emulated encoder output (EEO) | DRV.EMUSTEPCMD |
| 34A0h | | ARRAY | | | | PLS Position | |
| 34A0h | 0 | U8 | | RO | no | highest sub-index | — |
| 34A0h | 1 | INT32 | 1:1 | RW | no | Limit switch 1 compare value | PLS.P1 |
| 34A0h | 2 | INT32 | 1:1 | RW | no | Limit switch 2 compare value | PLS.P2 |
| 34A0h | 3 | INT32 | 1:1 | RW | no | Limit switch 3 compare value | PLS.P3 |
| 34A0h | 4 | INT32 | 1:1 | RW | no | Limit switch 4 compare value | PLS.P4 |
| 34A0h | 5 | INT32 | 1:1 | RW | no | Limit switch 5 compare value | PLS.P5 |
| 34A0h | 6 | INT32 | 1:1 | RW | no | Limit switch 6 compare value | PLS.P6 |
| 34A0h | 7 | INT32 | 1:1 | RW | no | Limit switch 7 compare value | PLS.P7 |
| 34A0h | 8 | INT32 | 1:1 | RW | no | Limit switch 8 compare value | PLS.P8 |
| 34A1h | | ARRAY | | | | PLS Width | — |
| 34A1h | 0 | U8 | | RO | no | highest sub-index | — |
| 34A1h | 1 | INT32 | 1:1 | RW | no | Sets Limit Switch1 Width | PLS.WIDTH1 |
| 34A1h | 2 | INT32 | 1:1 | RW | no | Sets Limit Switch 2 Width | PLS.WIDTH2 |
| 34A1h | 3 | INT32 | 1:1 | RW | no | Sets Limit Switch 3 Width | PLS.WIDTH3 |
| 34A1h | 4 | INT32 | 1:1 | RW | no | Sets Limit Switch 4 Width | PLS.WIDTH4 |
| 34A1h | 5 | INT32 | 1:1 | RW | no | Sets Limit Switch 5 Width | PLS.WIDTH5 |
| 34A1h | 6 | INT32 | 1:1 | RW | no | Sets Limit Switch 6 Width | PLS.WIDTH6 |
| 34A1h | 7 | INT32 | 1:1 | RW | no | Sets Limit Switch 7 Width | PLS.WIDTH7 |
| 34A1h | 8 | INT32 | 1:1 | RW | no | Sets Limit Switch 8 Width | PLS.WIDTH8 |
| 34A2h | | ARRAY | | | | PLS Time | — |
| 34A2h | 0 | U8 | | RO | no | highest sub-index | — |
| 34A2h | 1 | U16 | | RW | no | Sets limit switch 1 time | PLS.T1 |
| 34A2h | 2 | U16 | | RW | no | Sets limit switch 2 time | PLS.T2 |
| 34A2h | 3 | U16 | | RW | no | Sets limit switch 3 time | PLS.T3 |
| 34A2h | 4 | U16 | | RW | no | Sets limit switch 4 time | PLS.T4 |
| 34A2h | 5 | U16 | | RW | no | Sets limit switch 5 time | PLS.T5 |
| 34A2h | 6 | U16 | | RW | no | Sets limit switch 6 time | PLS.T6 |
| 34A2h | 7 | U16 | | RW | no | Sets limit switch 7 time | PLS.T7 |
| 34A2h | 8 | U16 | | RW | no | Sets limit switch 8 time | PLS.T8 |
| 34A3h | | ARRAY | | | | PLS Configuration | — |
| 34A3h | 0 | U8 | | RO | no | highest sub-index | — |
| 34A3h | 1 | U16 | | RW | no | Enables the limit switches | PLS.EN |
| 34A3h | 2 | U16 | | RW | no | Resets limit switches | PLS.RESET |
| 34A3h | 3 | U16 | | RW | no | Selects limit switch mode | PLS.MODE |
| 34A3h | 4 | U16 | | RW | no | Reads the limit switch state | PLS.STATE |
| 34A4h | 0 | U8 | | RW | no | Sets limit switch units | PLS.UNITS |
| 34A8h | 0 | INT32 | | RW | no | Sets the Compare 0 modulo value | CMP0.MODVALUE |
| 34A9h | | ARRAY | | | | Compare0 modulo bounds | — |
| 34A9h | 0 | U8 | | RO | no | highest sub-index | — |
| 34A9h | 1 | U8 | | RW | no | Compare0 modulo bound 1 | CMP0.MODBOUND1 |
| 34A9h | 2 | U8 | | RW | no | Compare0 modulo bound 2 | CMP0.MODBOUND2 |
| 34AAh | | ARRAY | | | | CMP0 setpoints | — |
| 34AAh | 0 | U8 | | RO | no | highest sub-index | — |
| 34AAh | 1 | INT32 | | RW | no | Compare0 setpoint 0 | CMP0.SETPOINT 0 |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|------------------|
| 34AAh | 2 | INT32 | | RW | no | Compare0 setpoint 1 | CMP0.SETPOINT 1 |
| 34AAh | 3 | INT32 | | RW | no | Compare0 setpoint 2 | CMP0.SETPOINT 2 |
| 34AAh | 4 | INT32 | | RW | no | Compare0 setpoint 3 | CMP0.SETPOINT 3 |
| 34AAh | 5 | INT32 | | RW | no | Compare0 setpoint 4 | CMP0.SETPOINT 4 |
| 34AAh | 6 | INT32 | | RW | no | Compare0 setpoint 5 | CMP0.SETPOINT 5 |
| 34AAh | 7 | INT32 | | RW | no | Compare0 setpoint 6 | CMP0.SETPOINT 6 |
| 34AAh | 8 | INT32 | | RW | no | Compare0 setpoint 7 | CMP0.SETPOINT 7 |
| 34ABh | | ARRAY | | | | CMP0 widths | — |
| 34ABh | 0 | U8 | | RO | no | highest sub-index | — |
| 34ABh | 1 | INT32 | | RW | no | Compare0 width 0 | CMP0.WIDTH 0 |
| 34ABh | 2 | INT32 | | RW | no | Compare0 width 1 | CMP0.WIDTH 1 |
| 34ABh | 3 | INT32 | | RW | no | Compare0 width 2 | CMP0.WIDTH 2 |
| 34ABh | 4 | INT32 | | RW | no | Compare0 width 3 | CMP0.WIDTH 3 |
| 34ABh | 5 | INT32 | | RW | no | Compare0 width 4 | CMP0.WIDTH 4 |
| 34ABh | 6 | INT32 | | RW | no | Compare0 width 5 | CMP0.WIDTH 5 |
| 34ABh | 7 | INT32 | | RW | no | Compare0 width 6 | CMP0.WIDTH 6 |
| 34ABh | 8 | INT32 | | RW | no | Compare0 width 7 | CMP0.WIDTH 7 |
| 34ACh | | ARRAY | | | | CMP0 widthtype | — |
| 34ACh | 0 | U8 | | RO | no | highest sub-index | — |
| 34ACh | 1 | U8 | | RW | no | Compare0 widthtype 0 | CMP0.WIDTHTYPE 0 |
| 34ACh | 2 | U8 | | RW | no | Compare0 widthtype 1 | CMP0.WIDTHTYPE 1 |
| 34ACh | 3 | U8 | | RW | no | Compare0 widthtype 2 | CMP0.WIDTHTYPE 2 |
| 34ACh | 4 | U8 | | RW | no | Compare0 widthtype 3 | CMP0.WIDTHTYPE 3 |
| 34ACh | 5 | U8 | | RW | no | Compare0 widthtype 4 | CMP0.WIDTHTYPE 4 |
| 34ACh | 6 | U8 | | RW | no | Compare0 widthtype 5 | CMP0.WIDTHTYPE 5 |
| 34ACh | 7 | U8 | | RW | no | Compare0 widthtype 6 | CMP0.WIDTHTYPE 6 |
| 34ACh | 8 | U8 | | RW | no | Compare0 widthtype 7 | CMP0.WIDTHTYPE 7 |
| 34ADh | | ARRAY | | | | CMP0 modes | — |
| 34ADh | 0 | U8 | | RO | no | highest sub-index | — |
| 34ADh | 1 | U8 | | RW | no | Compare0 mode 0 | CMP0.MODE 0 |
| 34ADh | 2 | U8 | | RW | no | Compare0 mode 1 | CMP0.MODE 1 |
| 34ADh | 3 | U8 | | RW | no | Compare0 mode 2 | CMP0.MODE 2 |
| 34ADh | 4 | U8 | | RW | no | Compare0 mode 3 | CMP0.MODE 3 |
| 34ADh | 5 | U8 | | RW | no | Compare0 mode 4 | CMP0.MODE 4 |
| 34ADh | 6 | U8 | | RW | no | Compare0 mode 5 | CMP0.MODE 5 |
| 34ADh | 7 | U8 | | RW | no | Compare0 mode 6 | CMP0.MODE 6 |
| 34ADh | 8 | U8 | | RW | no | Compare0 mode 7 | CMP0.MODE 7 |
| 34B0h | | ARRAY | | | | USER.DWORDS for writing of feedback memory | — |
| 34B0h | 0 | U8 | | RO | no | highest sub-index | — |
| 34B0h | 1 | U32 | | RW | no | FB1.USERDWORD1 | FB1.USERDWORD1 |
| 34B0h | 2 | U32 | | RW | no | FB1.USERDWORD2 | FB1.USERDWORD2 |
| 34B1h | | ARRAY | | | | USER.WORDS for writing of feedback memory | — |
| 34B1h | 0 | U8 | | RO | no | highest sub-index | — |
| 34B1h | 1 | U16 | | RW | no | FB1.USERWORD1 | FB1.USERWORD1 |
| 34B1h | 2 | U16 | | RW | no | FB1.USERWORD2 | FB1.USERWORD2 |
| 34B1h | 3 | U16 | | RW | no | FB1.USERWORD3 | FB1.USERWORD3 |
| 34B1h | 4 | U16 | | RW | no | FB1.USERWORD4 | FB1.USERWORD4 |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|------------------|
| 34B2h | | ARRAY | | | | USER.BYTES for writing of feedback memory | — |
| 34B2h | 0 | U8 | | RO | no | highest sub-index | — |
| 34B2h | 1 | U8 | | RW | no | FB1.USERBYTE1 | FB1.USERBYTE1 |
| 34B2h | 2 | U8 | | RW | no | FB1.USERBYTE2 | FB1.USERBYTE2 |
| 34B2h | 3 | U8 | | RW | no | FB1.USERBYTE3 | FB1.USERBYTE3 |
| 34B2h | 4 | U8 | | RW | no | FB1.USERBYTE4 | FB1.USERBYTE4 |
| 34B2h | 5 | U8 | | RW | no | FB1.USERBYTE5 | FB1.USERBYTE5 |
| 34B2h | 6 | U8 | | RW | no | FB1.USERBYTE6 | FB1.USERBYTE6 |
| 34B2h | 7 | U8 | | RW | no | FB1.USERBYTE7 | FB1.USERBYTE7 |
| 34B2h | 8 | U8 | | RW | no | FB1.USERBYTE8 | FB1.USERBYTE8 |
| 34B8h | 0 | INT32 | | RW | no | Sets the Compare 1 modulo value | CMP1.MODVALUE |
| 34B9h | | ARRAY | | | | Compare1 modulo bounds | — |
| 34B9h | 0 | U8 | | RO | no | highest sub-index | — |
| 34B9h | 1 | U8 | | RW | no | Compare1 modulo bound 1 | CMP1.MODBOUND1 |
| 34B9h | 2 | U8 | | RW | no | Compare1 modulo bound 2 | CMP1.MODBOUND2 |
| 34BAh | | ARRAY | | | | CMP1 setpoints | — |
| 34BAh | 0 | U8 | | RO | no | highest sub-index | — |
| 34BAh | 1 | INT32 | | RW | no | Compare1 setpoint 0 | CMP1.SETPOINT 0 |
| 34BAh | 2 | INT32 | | RW | no | Compare1 setpoint 1 | CMP1.SETPOINT 1 |
| 34BAh | 3 | INT32 | | RW | no | Compare1 setpoint 2 | CMP1.SETPOINT 2 |
| 34BAh | 4 | INT32 | | RW | no | Compare1 setpoint 3 | CMP1.SETPOINT 3 |
| 34BAh | 5 | INT32 | | RW | no | Compare1 setpoint 4 | CMP1.SETPOINT 4 |
| 34BAh | 6 | INT32 | | RW | no | Compare1 setpoint 5 | CMP1.SETPOINT 5 |
| 34BAh | 7 | INT32 | | RW | no | Compare1 setpoint 6 | CMP1.SETPOINT 6 |
| 34BAh | 8 | INT32 | | RW | no | Compare1 setpoint 7 | CMP1.SETPOINT 7 |
| 34BBh | | ARRAY | | | | CMP1 widths | — |
| 34BBh | 0 | U8 | | RO | no | highest sub-index | — |
| 34BBh | 1 | INT32 | | RW | no | Compare1 width 0 | CMP1.WIDTH 0 |
| 34BBh | 2 | INT32 | | RW | no | Compare1 width 1 | CMP1.WIDTH 1 |
| 34BBh | 3 | INT32 | | RW | no | Compare1 width 2 | CMP1.WIDTH 2 |
| 34BBh | 4 | INT32 | | RW | no | Compare1 width 3 | CMP1.WIDTH 3 |
| 34BBh | 5 | INT32 | | RW | no | Compare1 width 4 | CMP1.WIDTH 4 |
| 34BBh | 6 | INT32 | | RW | no | Compare1 width 5 | CMP1.WIDTH 5 |
| 34BBh | 7 | INT32 | | RW | no | Compare1 width 6 | CMP1.WIDTH 6 |
| 34BBh | 8 | INT32 | | RW | no | Compare1 width 7 | CMP1.WIDTH 7 |
| 34BCh | | ARRAY | | | | CMP1 widthtype | — |
| 34BCh | 0 | U8 | | RO | no | highest sub-index | — |
| 34BCh | 1 | U8 | | RW | no | Compare1 widthtype 0 | CMP1.WIDTHTYPE 0 |
| 34BCh | 2 | U8 | | RW | no | Compare1 widthtype 1 | CMP1.WIDTHTYPE 1 |
| 34BCh | 3 | U8 | | RW | no | Compare1 widthtype 2 | CMP1.WIDTHTYPE 2 |
| 34BCh | 4 | U8 | | RW | no | Compare1 widthtype 3 | CMP1.WIDTHTYPE 3 |
| 34BCh | 5 | U8 | | RW | no | Compare1 widthtype 4 | CMP1.WIDTHTYPE 4 |
| 34BCh | 6 | U8 | | RW | no | Compare1 widthtype 5 | CMP1.WIDTHTYPE 5 |
| 34BCh | 7 | U8 | | RW | no | Compare1 widthtype 6 | CMP1.WIDTHTYPE 6 |
| 34BCh | 8 | U8 | | RW | no | Compare1 widthtype 7 | CMP1.WIDTHTYPE 7 |
| 34BDh | | ARRAY | | | | CMP1 modes | — |
| 34BDh | 0 | U8 | | RO | no | highest sub-index | — |
| 34BDh | 1 | U8 | | RW | no | Compare1 mode 0 | CMP1.MODE 0 |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|--------------------------------------|
| 34BDh | 2 | U8 | | RW | no | Compare1 mode 1 | CMP1.MODE 1 |
| 34BDh | 3 | U8 | | RW | no | Compare1 mode 2 | CMP1.MODE 2 |
| 34BDh | 4 | U8 | | RW | no | Compare1 mode 3 | CMP1.MODE 3 |
| 34BDh | 5 | U8 | | RW | no | Compare1 mode 4 | CMP1.MODE 4 |
| 34BDh | 6 | U8 | | RW | no | Compare1 mode 5 | CMP1.MODE 5 |
| 34BDh | 7 | U8 | | RW | no | Compare1 mode 6 | CMP1.MODE 6 |
| 34BDh | 8 | U8 | | RW | no | Compare1 mode 7 | CMP1.MODE 7 |
| 34C0h | | ARRAY | | | | Compare0 handling | — |
| 34C0h | 0 | U8 | | RO | no | highest sub-index | — |
| 34C0h | 1 | U16 | | RW | no | Compare0 arm setpoints | CMP0.ARM 0..7 |
| 34C0h | 2 | U16 | | RW | no | Compare0 states | CMP0.STATE 0..7 |
| 34C1h | | ARRAY | | | | Compare1 handling | — |
| 34C1h | 0 | U8 | | RO | no | highest sub-index | — |
| 34C1h | 1 | U16 | | RW | no | Compare1 arm setpoints | CMP1.ARM 0..7 |
| 34C1h | 2 | U16 | | RW | no | Compare1 states | CMP1.STATE 0..7 |
| 3501h | 0 | INT32 | 1:1 | RW | no | Acceleration ramp | DRV.ACC, also see "6083h" (→ p. 996) |
| 3502h | 0 | INT32 | 1:1 | RW | no | Acceleration ramp for homing/jog modes | HOME.ACC |
| 3506h | 0 | INT32 | | | no | Action that hardware enable digital input will perform. | DRV.HWENMODE |
| 3509h | 0 | INT32 | 1000:1 | RO | no | Analog input voltage | AIN.VALUE |
| 3522h | 0 | INT32 | 1:1 | RW | no | Deceleration rate | DRV.DEC, also see "6084h" (→ p. 996) |
| 3524h | 0 | INT32 | 1:1 | RW | no | Deceleration ramp for homing/jog modes | HOME.DEC |
| 352Ah | 0 | INT32 | | RW | no | Direction of movements | DRV.DIR |
| 3533h | 0 | U32 | | RO | no | Resolution of motor encoder | FB1.ENCRES |
| 3534h | 0 | U32 | | RO | no | Mode of EEO connector | DRV.EMUEMODE |
| 3535h | 0 | U32 | | RO | no | Resolution of EEO | DRV.EMUERES |
| 3537h | 0 | U32 | | RO | no | Location of EEO index pulse | DRV.EMU EZOFFSET |
| 353Bh | 0 | INT32 | | RO | no | Selection of the feedback type | FB1.SELECT |
| 3542h | 0 | U32 | 1000:1 | RW | no | Position Control Loop: Proportional Gain | PL.KP |
| 3548h | 0 | U32 | 1000:1 | RW | no | Velocity Control Loop: Proportional Gain | VL.KP |
| 354Bh | 0 | INT32 | 1000:1 | RW | no | Sets the velocity loop velocity feedforward gain value | VL.KVFF |
| 354Dh | 0 | INT32 | 1000:1 | RW | no | Velocity Control Loop: I-Integration Time | VL.KI |
| 3558h | 0 | INT32 | 1000:1 | RO | no | Current Monitor | IL.FB |
| 3559h | 0 | INT32 | 1000:1 | RO | no | Drive Ifold | IL.DIFOLD |
| 355Ah | 0 | INT32 | 1000:1 | RW | no | I2T Warning | IL.FOLDWTHRESH |
| 3562h | 0 | INT32 | | RW | no | Function of Digital Input 1 | DIN1.MODE |
| 3565h | 0 | INT32 | | RW | no | Function of Digital Input 2 | DIN2.MODE |
| 3568h | 0 | INT32 | | RW | no | Function of Digital Input 3 | DIN3.MODE |
| 356Bh | 0 | INT32 | | RW | no | Function of Digital Input 4 | DIN4.MODE |
| 356Eh | 0 | INT32 | 1000:1 | RW | no | Application Peak Current, positive direction | IL.LIMITP |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|-----------------|
| 356Fh | 0 | INT32 | 1000:1 | RW | no | Application Peak Current, negative direction | IL.LIMITN |
| 3586h | 0 | U32 | | RW | no | Sets the motor temperature fault level | MOTOR.TEMPFAULT |
| 3587h | 0 | INT32 | | RW | no | Select Motor Holding Brake | MOTOR.BRAKE |
| 358Eh | 0 | U32 | 1000:1 | RW | no | Motor Continuous Current Rating | MOTOR.ICONT |
| 358Fh | 0 | U32 | 1000:1 | RW | no | Motor Peak Current Rating | MOTOR.IPEAK |
| 3593h | 0 | U32 | 1000:1 | RW | no | Sets the torque constant of the motor | MOTOR.KT |
| 3596h | 0 | U32 | 1000:1 | RO | no | Sets the proportional gain of the d-component current PI-regulator as a percentage of IL.KP | IL.KPDRATIO |
| 3598h | 0 | INT32 | 1000:1 | RW | no | Absolute Gain of Current Control loop | IL.KP |
| 359Ch | 0 | U32 | | RW | no | Sets the motor phase. | MOTOR.PHASE |
| 359Dh | 0 | U32 | | RW | no | Sets the number of motor poles | MOTOR.POLES |
| 35A3h | 0 | U32 | | RW | no | Sets the maximum motor speed | MOTOR.VMAX |
| 35A4h | 0 | INT32 | 1000:1 | RW | no | Maximum motor current | IL.MIFOLD |
| 35ABh | 0 | U32 | 1000:1 | RW | no | Sets the motor inertia | MOTOR.INERTIA |
| 35AFh | 0 | U32 | | RW | no | Sets the digital output 1 mode | DOOUT1.MODE |
| 35B2h | 0 | U32 | | RW | no | Sets the digital output 2 mode | DOOUT2.MODE |
| 35B4h | 0 | INT32 | | RW | no | Operating Mode | DRV.OPMODE |
| 35B8h | 0 | U32 | | RW | no | Table number for motion task | MT.TNUM |
| 35B9h | 0 | INT32 | | RW | no | Control for Motion Task 0 | MT.CNTL |
| 35BCh | 0 | INT32 | | RW | no | Next Task Number for Motion Task 0 | MT.MTNEXT |
| 35BDh | 0 | U32 | | RW | no | Time to next motion task | MT.TNEXT |
| 35C2h | 0 | INT32 | | RW | no | Select regen resistor | REGEN.REXT |
| 35C5h | 0 | INT32 | 1:1 | RO | no | Actual Following Error | PL.ERR |
| 35C6h | 0 | INT32 | 1:1 | RW | no | In-Position Window (profile position mode) | MT.TPOSWND |
| 35C7h | 0 | INT32 | 1:1 | RW | no | Max. Following Error | PL.ERRFTHRESH |
| 35CAh | 0 | INT32 | | RW | no | Position Resolution (Numerator) | UNIT.PIN |
| 35CBh | 0 | INT32 | | RW | no | Position Resolution (Denominator) | UNIT.POUT |
| 35CFh | 0 | INT32 | | RW | no | reserved | PL.MODPEN |
| 35D2h | 0 | U32 | | RO | no | Mechanical Position | FB1.MECHPOS |
| 35E2h | 0 | U32 | 1:1 | RW | no | Sets the current limit during homing procedure to a mechanical stop | HOME.IPEAK |
| 35EBh | 0 | INT32 | | WO | no | Save Data in EEPROM | DRV.NVSAVE |
| 35F0h | 0 | INT32 | | WO | no | Set Reference Point | HOME.SET |
| 35FEh | 0 | INT32 | | WO | no | Stop Motion Task | DRV.STOP |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|------------------|
| 35FFh | 0 | U32 | | RW | no | Selects between disable immediately or stop and then disable | DRV.DISMODE |
| 3610h | 0 | INT32 | | RO | no | Ambient Temperature | DRV.TEMPERATURES |
| 3611h | 0 | INT32 | | RO | no | Heat Sink Temperature | DRV.TEMPERATURES |
| 3612h | 0 | INT32 | | RO | no | Motor Temperature | MOTOR.TEMP |
| 3617h | 0 | U32 | 1:1 | RW | no | Undervoltage mode | VBUS.UVMODE |
| 3618h | 0 | INT32 | 1:1 | RO | no | Actual Velocity | VL.FB |
| 361Ah | 0 | INT32 | | RO | no | DC-bus voltage | VBUS.VALUE |
| 361Dh | 0 | U32 | 1000:1 | RW | no | Voltage level for undervoltage fault | VBUS.UVFTHRESH |
| 3622h | 0 | INT32 | 1:1 | RW | no | Max. Velocity | VL.LIMITP |
| 3623h | 0 | INT32 | 1:1 | RW | no | Max. Negative Velocity | VL.LIMITN |
| 3627h | 0 | INT32 | 1:1 | RW | no | Overspeed | VL.THRESH |
| 3629h | 0 | INT32 | 1000:1 | RW | no | SW1 Velocity Scaling Factor | AIN.VSCALE |
| 3637h | 0 | INT32 | 1:1 | RW | no | reserved | PL.MODP1 |
| 3638h | 0 | INT32 | 1:1 | RW | no | reserved | PL.MODP2 |
| 3656h | 0 | U64 | 1:1 | RW | no | Initial feedback position | FB1.ORIGIN |
| 3659h | 0 | INT32 | | RW | no | Type of acceleration setpoint for the system | UNIT.ACCROTARY |
| 365Bh | 0 | INT32 | | RW | no | Presetting for motion task that is processed later | MT.NUM |
| 365Fh | 0 | INT32 | | RW | no | Systemwide Definition of Velocity/Speed | UNIT.VROTARY |
| 3660h | 0 | INT32 | | RW | no | Set Resolution of the Position | UNIT.PROTARY |
| 366Eh | 0 | INT32 | | RW | no | Disable Delaytime with Holding Brake | MOTOR.TBRAKEAPP |
| 366Fh | 0 | INT32 | | RW | no | Enable Delaytime with Holding Brake | MOTOR.TBRAKERLS |
| 3683h | 0 | U16 | | RW | no | Delay for wake and shake timing | WS.TDELAY1 |
| 3685h | 0 | U16 | | RW | no | Sets delay for wake and shake timing | WS.TDELAY2 |
| 36D0h | 0 | U16 | | RW | no | Sets wake and shake current-vector appliance time | WS.T |
| 36D1h | 0 | U32 | 1:1 | RW | no | Sets the minimum movement required for wake and shake | WS.DISTMIN |
| 36D7h | 0 | U32 | 1000:1 | RW | no | Sets homing auto move flag | HOME.AUTOMOVE |
| 36E2h | 0 | U8 | | RW | no | Sets the number of repetitions for wake and shake | WS.NUMLOOPS |
| 36E5h | 0 | U32 | | RW | no | CAN baud rate selection | FBUS.PARAM01 |
| 36E6h | 0 | U32 | | RW | no | pll synchronization | FBUS.PARAM02 |
| 36E7h | 0 | U32 | | RW | no | - | FBUS.PARAM03 |
| 36E8h | 0 | U32 | | RW | no | SYNC surveillance | FBUS.PARAM04 |
| 36E9h | 0 | U32 | | RW | no | - | FBUS.PARAM05 |
| 36EAh | 0 | U32 | | RW | no | - | FBUS.PARAM06 |
| 36EBh | 0 | U32 | | RW | no | - | FBUS.PARAM07 |
| 36ECh | 0 | U32 | | RW | no | - | FBUS.PARAM08 |
| 36EDh | 0 | U32 | | RW | no | - | FBUS.PARAM09 |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|--------------|
| 36EEh | 0 | U32 | | RW | no | - | FBUS.PARAM10 |
| 36F6h | 0 | INT32 | | RW | no | Function of Digital Input 5 | DIN5.MODE |
| 36F9h | 0 | INT32 | | RW | no | Function of Digital Input 6 | DIN6.MODE |
| 36FCh | 0 | U32 | | RW | no | Function of Digital Input 7 | DIN7.MODE |
| 3856h | 0 | INT32 | 1:1 | RW | no | velocity window for profile position mode | MT.TVELWND |

Objects 5000h to 5999h

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|------------------|
| 5000h | 0 | UINT32 | | RW | no | Analog input low-pass filter cutoff frequency. | AIN.CUTOFF |
| 5001h | 0 | UINT32 | | RW | no | Analog input signal dead-band. | AIN.DEADBAND |
| 5002h | 0 | UINT32 | | RW | no | Analog current scale factor. | AIN.ISCALE |
| 5003h | 0 | UINT32 | | RW | no | Analog input offset. | AIN.OFFSET |
| 5009h | 0 | UINT32 | | RW | no | Analog current scale factor. | AOUT.ISCALE |
| 500Bh | 0 | UINT32 | | RW | no | Analog output offset. | AOUT.OFFSET |
| 5013h | 0 | UINT32 | | RW | no | Controls how often the excitation is updated. | BODE.EXCITEGAP |
| 5015h | 0 | UINT32 | | RW | no | Current command value used during the Bode procedure. | BODE.IAMP |
| 5016h | 0 | UINT32 | | RW | no | Sets whether the excitation uses current or velocity excitation type. | BODE.INJECTPOINT |
| 5019h | 0 | UINT32 | | RW | no | Length of the PRB signal before it repeats. | BODE.PRBDDEPTH |
| 5060h | 0 | UINT32 | | RW | no | Sets the fault relay mode. | DOUT.RELAYMODE |
| 5080h | 0 | UINT32 | | RW | no | Default state of the software enable. | DRV.ENDEFAULT |
| 5083h | 0 | UINT32 | | RW | no | Continuous rated current value. | DRV.ICONT |
| 5084h | 0 | UINT32 | | RW | no | Peak rated current value. | DRV.IPEAK |
| 5085h | 0 | UINT32 | | RW | no | Current that will be used during the DRV.ZERO procedure. | DRV.IZERO |
| 508Ch | 0 | UINT32 | | RW | no | Number of Biss Sensor (Position) Bits for the BiSS Mode C encoder in use. | FB1.BISSBITS |
| 508Fh | 0 | UINT32 | | RW | no | Initial feedback value as signed or unsigned. | FB1.INITSIGNED |
| 5096h | 0 | UINT32 | | RW | no | Current value used during the phase finding procedure (PFB.PFIND=1) | FB1.PFINDCMDU |
| 5097h | 0 | UINT32 | | RW | no | Number of feedback poles. | FB1.POLES |
| 5099h | 0 | UINT32 | | RW | no | Resolver nominal transformation ratio. | FB1.RESKTR |
| 509Ah | 0 | UINT32 | | RW | no | Electrical degrees of phase lag in the resolver. | FB1.RESREFPHASE |
| 509Ch | 0 | UINT32 | | RW | no | Controls tracking calibration algorithm. | FB1.TRACKINGCAL |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|----------------|
| 50B1h | 0 | UINT32 | | RW | no | Number of successful synchronized cycles needed to lock the PLL. | FBUS.PLLTHRESH |
| 50BBh | 0 | UINT32 | | RW | no | Denominator of the electronic gearing ratio; active in opmode 2 (position) only. | GEAR.IN |
| 50BCh | 0 | UINT32 | | RW | no | Electronic gearing mode; active in opmode 2 (position) only. | GEAR.MODE |
| 50BEh | 0 | UINT32 | | RW | no | Numerator of the electronic gearing ratio; active in opmode 2 (position) only. | GEAR.OUT |
| 50C5h | 0 | UINT32 | | RW | no | Homing direction | HOME.DIR |
| 50CBh | 0 | UINT32 | | RW | no | Homing mode | HOME.MODE |
| 50E2h | 0 | UINT32 | | RW | no | Current loops fieldbus injected feed-forward gain | IL.KBUSFF |
| 50FBh | 0 | UINT32 | | RW | no | Motor pitch. | MOTOR.PITCH |
| 50FEh | 0 | UINT32 | | RW | no | Type of thermal resistor inside the motor. | MOTOR.RTYPE |
| 5104h | 0 | UINT32 | | RW | no | Motor type. | MOTOR.TYPE |
| 510Eh | 0 | UINT32 | | RW | no | Motion task to be triggered after an emergency stop procedure; active in opmode 2 (position) only. | MT.EMERGMT |
| 5121h | 0 | UINT32 | | RW | no | Type of following error warning and fault usage. | PL.ERRMODE |
| 5128h | 0 | UINT32 | | RW | no | Feedback source for the position loop. | PL.FBSOURCE |
| 5175h | 0 | UINT32 | | RW | no | Service motion current 1; active in opmode 0 (torque) only. | SM.I1 |
| 5176h | 0 | UINT32 | | RW | no | Service motion current 2; active in opmode 0 (torque) only. | SM.I2 |
| 5177h | 0 | UINT32 | | RW | no | Service motion mode. | SM.MODE |
| 5179h | 0 | UINT32 | | RW | no | Service motion time 1. | SM.T1 |
| 517Ah | 0 | UINT32 | | RW | no | Service motion time 2. | SM.T2 |
| 517Eh | 0 | UINT32 | | RW | no | Enables and disables software travel limit switches. | SWLS.EN |
| 5184h | 0 | UINT32 | | RW | no | Linear acceleration/deceleration units. | UNIT.ACCLINEAR |
| 5187h | 0 | UINT32 | | RW | no | Linear position units. | UNIT.PLINEAR |
| 518Ah | 0 | UINT32 | | RW | no | Linear velocity units. | UNIT.VLINEAR |
| 518Eh | 0 | UINT32 | | RW | no | Voltage level for over voltage warning. | VBUS.OVWTHRESH |
| 51AEh | 0 | UINT32 | | RW | no | Feedback source for the velocity loop; active in opmodes 1 (velocity) and 2 (position) only. | VL.FBSOURCE |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|---------------------|
| 51B0h | 0 | UINT32 | | RW | no | Mode of velocity generation (Observer, d/dt); active in opmodes 1 (velocity) and 2 (position) only. | VL.GENMODE |
| 51B3h | 0 | UINT32 | | RW | no | Scales the observer velocity signal; active in opmodes 1 (velocity) and 2 (position) only. | VL.KO |
| 51B8h | 0 | UINT32 | | RW | no | Ratio of the estimated load moment of inertia relative to the motor moment of inertia; active in opmodes 1 and 2 only. | VL.LMJR |
| 51BAh | 0 | UINT32 | | RW | no | Bandwidth of the observer in Hz. | VL.OBSBW |
| 51BBh | 0 | UINT32 | | RW | no | Observer operating mode. | VL.OBSMODE |
| 51CBh | 0 | UINT32 | | RW | no | Filter mode for Digital In 1. | DIN1.FILTER |
| 51CCh | 0 | UINT32 | | RW | no | Filter mode for Digital In 2. | DIN2.FILTER |
| 51CDh | 0 | UINT32 | | RW | no | Filter mode for Digital In 3. | DIN3.FILTER |
| 51CEh | 0 | UINT32 | | RW | no | Filter mode for Digital In 4. | DIN4.FILTER |
| 51CFh | 0 | UINT32 | | RW | no | Filter mode for Digital In 5. | DIN5.FILTER |
| 51D0h | 0 | UINT32 | | RW | no | Filter mode for Digital In 6. | DIN6.FILTER |
| 51D1h | 0 | UINT32 | | RW | no | Filter mode for Digital In 7. | DIN7.FILTER |
| 51E7h | 0 | UINT32 | | RW | no | Modbus User Units Input parameter | MODBUS.PIN |
| 51E8h | 0 | UINT32 | | RW | no | Modbus User Units Output parameter. | MODBUS.POUT |
| 51E9h | 0 | UINT32 | | RW | no | Feedback Resolution (per rev) over Modbus. | MODBUS.PSCALE |
| 51ECh | 0 | UINT32 | | RW | no | Secondary feedback (FB2) resolution. | FB2.ENCRESES |
| 51EDh | 0 | UINT32 | | RW | no | Mode for the second feedback inputs and high speed digital inputs. | FB2.MODE |
| 51EEh | 0 | UINT32 | | RW | no | Source for the second feedback input. | FB2.SOURCE |
| 51EFh | 0 | UINT32 | | RW | no | Brake apply timeout for vertical axis. | MOTOR.TBRAKETO |
| 51F0h | 0 | UINT32 | | RW | no | i.p. | MODBUS.MSGLOG |
| 520Ch | 0 | UINT32 | | RW | no | Scaling mode for Modbus values. | MODBUS.SCALING |
| 520Dh | 0 | UINT32 | | RW | no | Encoder output pulse width for modes 6 to 7. | DRV.EMUEPULSE-WIDTH |
| 520Eh | 0 | UINT32 | | RW | no | Enable/disable motor velocity vs. maximum emulated encoder velocity monitoring function. | DRV.EMUECHECK-SPEED |
| 5251h | 0 | UINT32 | | RW | no | Analog input deadband mode. | AIN.DEADBANDMODE |
| 5252h | 0 | UINT32 | | RW | no | Analog input mode | AIN.MODE |
| 5253h | 0 | UINT32 | | RW | no | Direction of IOs from X9. | DIO10.DIR |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|------------------|
| 5254h | 0 | UINT32 | | RW | no | Inverting the output voltage of the IO, when in the output direction. | DIO10.INV |
| 5255h | 0 | UINT32 | | RW | no | Direction of IOs from X9. | DIO11.DIR |
| 5256h | 0 | UINT32 | | RW | no | Inverting the output voltage of the IO, when in the output direction. | DIO11.INV |
| 5257h | 0 | UINT32 | | RW | no | Direction of IOs from X9. | DIO9.DIR |
| 5258h | 0 | UINT32 | | RW | no | Inverting the output voltage of the IO, when in the output direction. | DIO9.INV |
| 5259h | 0 | UINT32 | | RW | no | Fault Action for Fault 130. | FAULT130.ACTION |
| 525Ah | 0 | UINT32 | | RW | no | Fault Action for Fault 131. | FAULT131.ACTION |
| 525Bh | 0 | UINT32 | | RW | no | Fault Action for Fault 132. | FAULT132.ACTION |
| 525Ch | 0 | UINT32 | | RW | no | Fault Action for Fault 133. | FAULT134.ACTION |
| 525Dh | 0 | UINT32 | | RW | no | Fault Action for Fault 702. | FAULT702.ACTION |
| 525Eh | 0 | UINT32 | | RW | no | Method of acquiring IP Address. | IP.MODE |
| 525Fh | 0 | UINT32 | | RW | no | Load inertia. | LOAD.INERTIA |
| 5260h | 0 | UINT32 | | RW | no | Motor back EMF constant. | MOTOR.KE |
| 5261h | 0 | UINT32 | | RW | no | Changing voltage thresholds. | VBUS.HALFVOLT |
| 5262h | 0 | UINT32 | | RW | no | Direction for the second feedback input (X9 and X7). | FB2.DIR |
| 5263h | 0 | UINT32 | | RW | no | Feedback for handwheel operation. | DRV.HANDWHEELSRC |
| 5264h | 0 | UINT32 | | RW | no | Delay time between inactive Hardware Enable input and drive disable. | DRV.HWENDELAY |
| 5265h | 0 | UINT32 | | RW | no | Index into the Current Loop Gain Scheduling Table. | IL.KPLOOKUPINDEX |
| 5266h | 0 | UINT32 | | RW | no | Value of the current loop gain scheduling index. | IL.KPLOOKUPVALUE |
| 5267h | 0 | UINT32 | | RW | no | Fault Action for Fault 451. | FAULT451.ACTION |
| 5268h | 0 | UINT32 | | RW | no | Brake Immediately in the case of a drive disable. | MOTOR.BRAKEIMM |
| 5352h | 0 | UINT16 | | RW | no | Amount of time a communication error must be present before an W&S-fault is thrown. | WS.CHECKT |
| 535Ch | 0 | UINT16 | | RW | no | Sets the calming time of the motor for Wake & Shake mode 1. | WS.TSTANDSTILL |
| 535Dh | 0 | UINT16 | | RW | no | Time for the ramp up current in Wake & Shake mode 1. | WS.TIRAMP |
| 5360h | 0 | UINT16 | | RW | no | Rotor time constant. | MOTOR.IMTR |
| 5361h | 0 | UINT8 | | RW | no | Sets the feedback source for the current loop for MOTOR.TYPE4. | IL.FBSOURCE |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|----------------------|
| 5362h | 0 | UINT32 | | RW | no | The direct-axis current set point used for induction machine closed-loop control. | MOTOR.IMID |
| 538Bh | 0 | UINT16 | | RW | no | | DRV.EMUESTEPMODE () |
| 538Ch | 0 | UINT16 | | RW | no | | DRV.EMUESTEPSTATUS |
| 538Dh | 0 | UINT16 | | RW | no | | DRV.EMUESTEPVMAX |
| 538Fh | 0 | INT8 | | RW | no | Compare engine 0 source | CMP0.SOURCE |
| 5390h | 0 | INT8 | | RW | no | Compare engine 1 source | CMP1.SOURCE |
| 5394h | 0 | U16 | | RW | no | Compare engine 0 output mask | CMP0.OUTMASK |
| 539Bh | 0 | U16 | | RW | no | Compare engine 1 output mask | CMP1.OUTMASK |
| 53A6h | 0 | U8 | | RW | no | Compare engine 0 modulo enable | CMP0.MODEN |
| 53ADh | 0 | U8 | | RW | no | Compare engine 1 modulo enable | CMP1.MODEN |
| 53B1h | 0 | U32 | | RW | no | Compare engine 0 advance | CMP0.ADVANCE |
| 53B2h | 0 | U32 | | RW | no | Compare engine 1 advance | CMP1.ADVANCE |
| 53C7h | 0 | UINT32 | | RW | no | Sets the fault display mode | DRV.FAULTDISPLAYMODE |
| 53D5h | 0 | UINT32 | | RW | no | Sets the delay time for PL.CMD | PL.PDELAY |
| 53D6h | 0 | UINT32 | | RW | no | Sets the delay time for the velocity feedforward integrator component | VL.FFDELAY |
| 53D7h | 0 | INT8 | | RW | no | Allows a surface permanent magnet motor to operate as an interior permanent magnet motor | MOTOR.FIELDWEAKENING |
| 5403h | 0 | UINT32 | | RW | no | Toggles between HOME.IPEAK and current loop limits during homing | HOME.IPEAKACTIVE |
| 5404h | 0 | UINT32 | | RW | no | Scaling factor (numerator) for the command DRV.EMUESTEPCMD | DRV.EMUESTEPCMDPIN |
| 5405h | 0 | UINT32 | | RW | no | Scaling factor (denominator) for the command DRV.EMUESTEPCMD | DRV.EMUESTEPCMDPOUT |
| 5406h | 0 | UINT32 | | RW | no | Sets the target position window for the homing procedure; active in opmode 2 (position) only | HOME.TPOSWND |

19.33.2.5 Profile specific SDOs

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|------------------------|--------------|
| 6040h | 0 | U16 | | WO | yes | Control word | — |
| 6041h | 0 | U16 | | RO | yes | Status word | — |
| 605Ah | 0 | INT16 | | RW | no | Quick stop option code | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|------------------|
| 6060h | 0 | INT8 | | RW | yes | Modes of Operation | — |
| 6061h | 0 | INT8 | | RO | yes | Modes of Operation Display | — |
| 6063h | 0 | INT32 | | RO | yes | Position actual value (increments) | — |
| 6064h | 0 | INT32 | 1:1 | RO | yes | Position actual value (position units) | PL.FB |
| 6065h | 0 | U32 | 1:1 | RW | no | Following error window | PL.ERRFTHRESH |
| 606Bh | 0 | INT32 | 1:1 | RO | no | Velocity demand value | VL.CMD |
| 606Ch | 0 | INT32 | 1000:1 | RO | yes | Velocity actual value (PDO in RPM) | VL.FB |
| 606Dh | 0 | U16 | | RW | yes | Velocity window | |
| 606Eh | 0 | U16 | | RW | yes | Velocity window time | |
| 6071h | 0 | INT16 | | RW | yes* | Target torque | — |
| 6072h | 0 | U16 | | RW | yes* | Max torque | — |
| 6073h | 0 | U16 | | RW | no | Max current | |
| 6077h | 0 | INT16 | | RO | yes | Torque actual value | DRV.ICONT |
| 607Ah | 0 | INT32 | 1:1 | RW | yes | Target position | MT.P |
| 607Ch | 0 | INT32 | 1:1 | RW | no | Reference offset | HOME.P |
| 607Dh | | ARRAY | | | | Software position limit | |
| 607Dh | 0 | U8 | | RO | no | highest sub-index | |
| 607Dh | 1 | INT32 | 1:1 | RW | no | Software position limit 1 | SWLS.LIMIT0 |
| 607Dh | 2 | INT32 | 1:1 | RW | no | Software position limit 2 | SWLS.LIMIT1 |
| 6081h | 0 | U32 | 1:1 | RW | yes | Profile Velocity | MT.V |
| 6083h | 0 | U32 | 1:1 | RW | yes | Profile Acceleration | MT.ACC , DRV.ACC |
| 6084h | 0 | U32 | 1:1 | RW | yes | Profile Deceleration | MT.DEC , DRV.DEC |
| 6087h | 0 | U32 | | RW | yes | Torque slope | — |
| 608Fh | | ARRAY | | | | Position encoder resolution | — |
| 608Fh | 0 | U8 | | RO | no | highest sub-index | — |
| 608Fh | 1 | U32 | | RW | no | Encoder increments | — |
| 608Fh | 2 | U32 | | RW | no | Motor revolutions | |
| 6091h | | ARRAY | | | | Gear ratio | — |
| 6091h | 0 | U8 | | RO | no | highest sub-index | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|----------------------|
| 6091h | 1 | U32 | | RW | no | Motor revolution | |
| 6091h | 2 | U32 | | RW | no | Shaft revolutions | |
| 6092h | | ARRAY | | | | Feed constant | — |
| 6092h | 0 | U8 | | RO | no | highest sub-index | — |
| 6092h | 1 | U32 | | RW | no | Feed | UNIT.PIN |
| 6092h | 2 | U32 | | RW | no | Shaft revolutions | — |
| 6098h | 0 | INT8 | | RW | no | Homing type | HOME.MODE , HOME.DIR |
| 6099h | | ARRAY | | | | Homing velocity | — |
| 6099h | 0 | U8 | | RO | no | highest sub-index | — |
| 6099h | 1 | U32 | 1:1 | RW | no | Speed while searching for limit switch | HOME.V |
| 6099h | 2 | U32 | | RW | no | Speed while searching for zero mark | HOME.FEEDRATE |
| 609Ah | 0 | U32 | 1:1 | RW | no | Homing acceleration | HOME.ACC , HOME.DEC |
| 60B1h | 0 | INT32 | 1:1 | RW | yes* | Velocity offset | VL.BUSFF |
| 60B2h | 0 | INT16 | | RW | yes* | Torque offset (PDO only) | |
| 60B8h | 0 | U16 | | RW | yes | Touch probe function | — |
| 60B9h | 0 | U16 | | RW | yes | Touch probe status | — |
| 60BAh | 0 | INT32 | | RW | yes | Touch probe 1 positive edge | — |
| 60BBh | 0 | INT32 | | RW | yes | Touch probe 1 negative edge | — |
| 60BCh | 0 | INT32 | | RW | yes | Touch probe 2 positive edge | — |
| 60BDh | 0 | INT32 | | RW | yes | Touch probe 2 negative edge | — |
| 60C0h | 0 | INT16 | | RW | no | Interpolation sub-mode select | — |
| 60C1h | | ARRAY | | | | Interpolation data record | — |
| 60C1h | 0 | U8 | | RO | no | highest sub-index | — |
| 60C1h | 1 | INT32 | | RW | yes* | Interpolation target position | — |
| 60C1h | 2 | U32 | | RW | yes | Interpolation time | — |
| 60C1h | 3 | INT32 | | RW | yes | Interpolation target velocity | — |
| 60C2h | | RECORD | | | | Interpolation time period | — |
| 60C2h | 0 | U8 | | RO | no | highest sub-index | FBUS.SAMPLEPERIOD |
| 60C2h | 1 | U8 | | RW | no | Interpolation time units | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|---------------------------|
| 60C2h | 2 | INT8 | | RW | no | Interpolation time index | — |
| 60C4h | | RECORD | | | | Interpolation data configuration | — |
| 60C4h | 0 | U8 | | RO | no | highest sub-index | — |
| 60C4h | 1 | U32 | | RO | no | Maximum buffer size | — |
| 60C4h | 2 | U32 | | RO | yes | Actual buffer size | — |
| 60C4h | 3 | U8 | | RW | no | Buffer organization | — |
| 60C4h | 4 | U16 | | RW | no | Buffer position | — |
| 60C4h | 5 | U8 | | WO | no | Size of data record | — |
| 60C4h | 6 | U8 | | WO | no | Buffer clear | — |
| 60D0h | | ARRAY | | | | Touch probe source | — |
| 60D0h | 0 | U8 | | RO | no | highest sub-index | - |
| 60D0h | 1 | INT16 | | RW | no | Touch probe 1 source | — |
| 60D0h | 2 | INT16 | | RW | no | Touch probe 2 source | — |
| 60E0h | 0 | UINT16 | | RO | yes* | Positive torque limit value | IL.LIMITP |
| 60E1h | 0 | UINT16 | | RO | yes* | Negative torque limit value | IL.LIMITN |
| 60E4h | | ARRAY | | | | Additional position actual value | — |
| 60E4h | 0 | U8 | | RO | no | highest sub-index | — |
| 60E4h | 1 | INT32 | | RW | no | 1st additional position actual value | — |
| 60E4h | 2 | INT32 | | RW | no | reserved | — |
| 60E4h | 3 | INT32 | | RW | no | 3rd additional position actual value | — |
| 60E8h | | ARRAY | | | | Additional gear ratio - motor shaft revolutions | — |
| 60E8h | 0 | U8 | | RO | no | highest sub-index | — |
| 60E8h | 1 | U32 | | RW | no | 1st additional gear ratio - motor shaft revolutions | — |
| 60E8h | 2 | U32 | | RW | no | reserved | — |
| 60E8h | 3 | U32 | | RW | no | 3rd additional gear ratio - motor shaft revolutions | DS402.3ADDPOSGEARMOTORREV |
| 60E9h | | ARRAY | | | | Additional feed constant - feed | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|--|
| 60E9h | 0 | U8 | | RO | no | highest sub-index | — |
| 60E9h | 1 | U32 | | RW | no | 1st additional feed constant - feed | DS402.1ADDPOSFCFEED |
| 60E9h | 2 | U32 | | RW | no | reserved | — |
| 60E9h | 3 | U32 | | RW | no | 3rd additional feed constant - feed | DS402.3ADDPOSFCFEED |
| 60EDh | | ARRAY | | | | Additional gear ratio - driving shaft revolutions | — |
| 60EDh | 0 | U8 | | RO | no | highest sub-index | — |
| 60EDh | 1 | U32 | | RW | no | 1st additional gear ratio - driving shaft revolutions | DS402.1ADDPOSGEARSHAFTREV |
| 60EDh | 2 | U32 | | RW | no | reserved | — |
| 60EDh | 3 | U32 | | RW | no | 3rd additional gear ratio - driving shaft revolutions | DS402.3ADDPOSGEARSHAFTREV |
| 60EEh | | ARRAY | | | | Additional feed constant - driving shaft revolutions | — |
| 60EEh | 0 | U8 | | RO | no | highest sub-index | — |
| 60EEh | 1 | U32 | | RW | no | 1st additional feed constant - driving shaft revolutions | DS402.1ADDPOSFCFSHAFTREV |
| 60EEh | 2 | U32 | | RW | no | reserved | — |
| 60EEh | 3 | U32 | | RW | no | 3rd additional feed constant - driving shaft revolutions | DS402.3ADDPOSFCFSHAFTREV |
| 60F4h | 0 | INT32 | | RO | yes | Following error actual value | PL.ERR |
| 60FCh | 0 | INT32 | | RO | yes | Position demand internal value | PL.CMD |
| 60FDh | 0 | U32 | | RO | yes | Digital inputs | DIN1.MODE TO DIN6.MODE DIN1.MODE TO DIN3.MODE |
| 60FEh | | ARRAY | | | | Digital outputs | |
| 60FEh | 0 | U8 | | RO | no | highest sub-index | — |
| 60FEh | 1 | U32 | | RW | yes | Physical outputs | |
| 60FEh | 2 | U32 | | RW | no | Bit mask | |
| 60FFh | 0 | INT32 | 1000:1 | RW | yes* | Target velocity | VL.CMDU |
| 6502h | 0 | U32 | | RO | no | Supported drive modes | — |

19.33.3 Object descriptions

The objects in this section are sorted by object number.

19.33.3.1 Object 1000h: Device Type (DS301)

This object describes the device type (servo drive) and device functionality (DS402 drive profile). Definition:

| | | | | | | | |
|------------------------|----|------|----|-----------------------|--|--|---|
| MSB | | | | LSB | | | |
| Additional information | | | | Device profile number | | | |
| Mode bits | | Type | | 402d=192h | | | |
| 31 | 24 | 23 | 16 | 15 | | | 0 |

The device profile number is DS402, the type is 2 for drives, the mode bits 28 to 31 are manufacturer specific and may be changed from its actual value of 0. A read access delivers 0x00020192 at the moment.

| | |
|---------------|--------------|
| Index | 1000h |
| Name | device type |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | no |

19.33.3.2 Object 1001h: Error register (DS301)

This object is an error register for the device. The device can map internal errors into this byte. It is a part of an Emergency object.

| | |
|---------------|----------------|
| Index | 1001h |
| Name | Error register |
| Object code | VAR |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | no |

Error reasons to be signaled: If a bit is set to 1 the specified error has occurred. The generic error is signaled at any error situation.

| Bit | Description | Bit | Description |
|-----|---------------|-----|--|
| 0 | generic error | 4 | communication error (overrun, error state) |
| 1 | current | 5 | device profile specific |
| 2 | voltage | 6 | reserved (always 0) |
| 3 | temperature | 7 | manufacturer specific |

19.33.3.3 Object 1002h: Manufacturer Status Register (DS301)

The manufacturer status register contains important drive informations.

| | |
|---------------|------------------------------|
| Index | 1002h |
| Name | Manufacturer Status Register |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | UNSIGNED32 |
| Default value | no |

The following table shows the bit assignment for the status register:

| Bit | Description | Bit | Description |
|-----|--|-----|---------------------------------------|
| 0 | 1 = Movement (positioning, homing) active | 16 | 1 = Homing move active |
| 1 | reference position set | 17 | reserved |
| 2 | 1 = reference switch high (home-position) | 18 | reserved |
| 3 | 1 = In Position | 19 | 1 = Emergency stop active |
| 4 | reserved | 20 | reserved |
| 5 | reserved | 21 | reserved |
| 6 | reserved | 22 | reserved |
| 7 | Active Disabel activated | 23 | 1 = Homing move finished |
| 8 | Warning active | 24 | Power stage deactivating |
| 9 | 1 = target velocity reached (pp- or pv-Mode) | 25 | 1 = digital input 1 set |
| 10 | reserved | 26 | 1 = digital input 2 set |
| 11 | 1 = Homing error | 27 | 1 = digital input 3 set |
| 12 | reserved | 28 | 1 = digital input 4 set |
| 13 | 1 = Safe Torque Off selected | 29 | 1 = digital input hardware enable set |
| 14 | 1 = Power stage enabled | | |
| 15 | 1 = Error state | 31 | Braking, 1 = set points not accepted |

19.33.3.4 Object 1003h: Predefined Error Field (DS301)

The object 1003h provides an error history with a maximum size of 10 entries.

Subindex 0 contains the number of errors which have occurred since the last reset of the error history, either by startup of the drive or resetting the error history by writing 0 to subindex 0.

A new Emergency-message is written into subindex 1 shifting the old entries one subindex higher. The old content of subindex 8 is lost.

The UNSIGNED32-information written to the subindexes is defined in the field Error Code in the description of the Emergency Messages (→ p. 968).

| | |
|---------------|---------------------------------|
| Index | 1003h |
| Name | pre-defined Error Field |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | 0 to 10 |
| Default value | 0 |
| Subindex | 1 to 10 |
| Description | Standard error field (→ p. 968) |
| Category | optional |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | no |

19.33.3.5 Object 1005h: COB-ID of the SYNC Message (DS301)

This object defines the COB-Id of the synchronisation object (SYNC).

| | |
|---------------|-----------------------------|
| Index | 1005h |
| Name | COB-ID for the SYNC message |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | conditional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0x80 |

Bit coded information:

| Bit | Value | Meaning |
|---------------|-------|----------------------------------|
| 31 (MSB) | X | — |
| 30 | 0 | Device not generate SYNC message |
| | 1 | Device generates SYNC message |
| 29 | 0 | 11 Bit ID (CAN 2.0A) |
| | 1 | 29 Bit ID (CAN 2.0B) |
| 28 to 11 | X | — |
| | 0 | if Bit 29=0 |
| 10 to 0 (LSB) | X | Bit 0 to 10 of SYNC COB-ID |

The device does not support the generation of SYNC-messages and only the 11-bit IDs. So the bits 11 to 30 are always 0.

19.33.3.6 Object 1006h: Communication Cycle Period (DS301)

This object can be used to define the period (in μ s) for the transmission of the SYNC telegram.

| | |
|---------------|-----------------------------------|
| Index | 1006h |
| Name | Period of the communication cycle |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | O |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 00h |

19.33.3.7 Object 1008h: Manufacturer Device Name (DS301)

The device name consists of four ASCII characters in the form Yzzz, whereby Y stands for the mains voltage (L, M, H or U, e.g. H for High Voltage) zzz stands for the power stage current.

| | |
|---------------|--------------------------|
| Index | 1008h |
| Name | Manufacturer Device Name |
| Object code | VAR |
| Data type | Visible String |
| Category | Optional |
| Access | const |
| PDO mapping | not possible |
| Value range | |
| Default value | no |

19.33.3.8 Object 1009h: Manufacturer Hardware Version

This object will be supported in the future.

| | |
|---------------|-------------------------------|
| Index | 1009h |
| Name | manufacturer hardware version |
| Object code | VAR |
| Data type | Visible String |
| Category | Optional |
| Access | const |
| PDO mapping | not possible |
| Value range | - |
| Default value | no |

19.33.3.9 Object 100Ah: Manufacturer Software Version (DS301)

The object contains the manufacturer software version (here: the CANopen-part of the drive firmware).

| | |
|---------------|-------------------------------|
| Index | 100Ah |
| Name | Manufacturer Software Version |
| Object code | VAR |
| Data type | Visible String |
| Category | Optional |
| Access | const |
| PDO mapping | not possible |
| Value range | 0.01 to 9.99 |
| Default value | no |

19.33.3.10 Object 100Ch: Guard Time (DS301)Response monitoring

The arithmetical product of the Objects 100Ch Guard Time and 100Dh Lifetime Factor is the response monitoring time. The Guard Time is given in milliseconds. The response monitoring is activated with the first Nodeguard object. If the value of the object Guard Time is set to zero, then the response monitoring is inactive.

| | |
|---------------|--|
| Index | 100Ch |
| Name | Guard Time |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | conditional; mandatory, if heartbeat not supported |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED16 |
| Default value | 0 |

19.33.3.11 Object 100Dh: Lifetime Factor (DS301)

The product of Guard Time and Life Time Factor gives the life time for the nodeguarding protocol. If it's 0, the protocol is not used.

| | |
|---------------|--|
| Index | 100Dh |
| Name | Lifetime Factor |
| Object code | VAR |
| Data type | UNSIGNED8 |
| Category | conditional; (mandatory, if heartbeat not supported) |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | 0 |

19.33.3.12 Object 1010h: Store Parameters (DS301)

This object supports the saving of parameters to a flash EEPROM. Only the subindex 1 for saving of all parameters, which can also be saved in the parameter files via the GUI, is supported.

| | |
|---------------|-------------------------------|
| Index | 1010h |
| Name | store parameters (DRV.NVSAVE) |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Name | highest sub-index supported |
| Object code | VAR |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value range | 1 |
| Default value | 1 |
| Subindex | 1 |
| Name | save all parameters |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |

Data definition:

| Bit | Value | Meaning |
|---------|-------|--|
| 31 to 2 | 0 | reserved (=0) |
| 1 | 0 | Device does not save parameters autonomously |
| | 1 | Device does save parameters autonomously |
| 0 | 0 | Device does not save parameters on command |
| | 1 | Device does save parameters on command |

By read access to subindex 1 the drive provides information about its storage functionality.

This drive provides a constant value of 1 by read access, i.e. all parameters can be saved by writing to Object 1010 sub 1. In general the drive does not save parameters autonomously with the exception of e.g. the special treatment of the homing of multiturn absolute encoders.

Storing of parameters is only done if a special signature ("save") is written to subindex 1. "save" is equivalent to the unsigned32 - number 65766173h.

19.33.3.13 Object 1011h: Restore Default Parameters DS301

With this object the default values of parameters according to the communication or device profile are restored. The AKD PDMM gives the possibility to restore all default values.

| | |
|---------------|---|
| Index | 1011h |
| Name | restore default parameters |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Name | highest sub-index supported |
| Object code | VAR |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value range | 1 |
| Default value | 1 |
| Subindex | 1 |
| Name | restore all default parameters (DRV.RSTVAR) |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 (device restores parameter) |

Restoring default parameters to the RAM will be done, if a special signature ("load") is written to subindex 1. "load" has to be transmitted as unsigned32 - number 64616F6Ch.

19.33.3.14 Object 1012h: COB-ID of the Time Stamp (DS301)

This object defines the COB-ID of the time stamp.

| | |
|---------------|---------------------------|
| Index | 1012h |
| Name | COB-ID for the time stamp |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 100h |

Bit coded information:

| Bit | Content | Value | Meaning |
|---------------|----------|-----------|-------------------------------------|
| 31 (MSB) | consume | 0 | Drive does not consume time message |
| | | 1 | Drive does consume time message |
| 30 | produce | 0 | Drive does not produce time message |
| | | 1 | Drive does produce time message |
| 29 | frame | 0 | Value fixed to 0 |
| 28 to 11 | reserved | – | reserved |
| 10 to 0 (LSB) | CAN-ID | 0h - 800h | COB-ID of the time stamp |

19.33.3.15 Object 1014h: COB-ID for Emergency Message (DS301)

This object defines the COB-ID of the Emergency message.

| | |
|---------------|---|
| Index | 1014h |
| Name | COB-ID emergency message |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | conditional; mandatory, if Emergency is supported |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 80h + Node - ID |

19.33.3.16 Object 1016h: Consumer Heartbeat Time

The consumer heartbeat time defines the expected heartbeat cycle time (ms) and must be higher than the corresponding producer heartbeat time configured on the device producing this heartbeat. Monitoring starts after the reception of the first heartbeat. If the consumer heartbeat time is 0 ms the corresponding entry is not used.

| | |
|---------------|-----------------------------|
| Index | 1016h |
| Name | consumer heartbeat time |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value range | 1 |
| Default value | 1 |
| Subindex | 1 |
| Description | Consumer heartbeat time |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value range | unsigned 32 |
| Default value | 0 |

Definition of the entry value of Subindex 1

| | MSB | | | LSB | |
|------------|----------------------|----|-----------|-----|----------------|
| Value | reserved (value: 00) | | Node-ID | | heartbeat time |
| Encoded as | - | | UNSIGNED8 | | UNSIGNED16 |
| Bit | 31 | 24 | 23 | 16 | 15 0 |

19.33.3.17 Object 1017h: Producer Heartbeat Time

The producer heartbeat time defines the cycle time of the heartbeat in ms. If it's 0, it is not used.

| | |
|---------------|---|
| Index | 1017h |
| Name | Producer heartbeat time |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | conditional; mandatory, if guarding is not supported |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED16 |
| Default value | 0 |

19.33.3.18 Object 1018h: Identity Object (DS301)

The Identity Object contains general device information.

| | |
|---------------|-----------------------------|
| Index | 1018h |
| Name | Identity Object |
| Object code | RECORD |
| Data type | Identity |
| Category | mandatory |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 1 to 4 |
| Default value | 4 |

Subindex 1 is a unique number for a device manufacturer.

| | |
|---------------|--------------------|
| Subindex | 1 |
| Description | Vendor ID |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0x6Ah (Kollmorgen) |

Subindex 2 contains four ASCII - characters, which determine the voltage range and current class of the device. The voltage range is one character L, M or H for low, medium and high voltage. The next three characters are showing the continuous current of the drive.

| | |
|---------------|----------------------------|
| Subindex | 2 |
| Description | Product Code |
| Category | optional |
| Access | R/O |
| PDO mapping | not possible |
| Value range | e.g. M006 for an MV6 drive |
| Default value | no |

Subindex 3 consists of two revision numbers:

- the major revision number in the upper word containing the CAN-version
- the minor revision number is not used in the AKD PDMM. The firmware version can be retrieved as a string via object 0x100A or as numbers via object 0x2018 subindex 1 to 4.

E.g. a value of 0x0014 0000 means CAN-version 0.20.

| | |
|---------------|-----------------|
| Subindex | 3 |
| Description | Revision Number |
| Category | optional |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | no |

Subindex 4 gives the serial number of the drive. This number contains the following information in it:

- bits 0..14: Board serial number (production in week of year)
- bits 15..20: week of production
- bits 21..24: year of production - 2009
- bits 25..31: ASCII-code of MFR-ID

| | |
|---------------|---------------|
| Subindex | 4 |
| Description | Serial Number |
| Category | optional |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | no |

19.33.3.19 Object 1026h: OS Prompt

The OS prompt is used to build up an ASCII - communication channel to the drive.

| | |
|---------------|-----------------------------|
| Index | 1026h |
| Name | OS Prompt |
| Object code | ARRAY |
| Data type | UNSIGNED8 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | 2 |

Subindex 1 is used to send one character to the drive.

| | |
|---------------|--------------|
| Subindex | 1 |
| Description | StdIn |
| Category | mandatory |
| Access | W |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | — |

Subindex 2 is used to receive one character from the drive.

| | |
|---------------|--------------|
| Subindex | 2 |
| Description | StdOut |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | 0 |

19.33.3.20 Objects 1400-1403h: 1st - 4th RxPDO communication parameter (DS301)

1400h to 1403h for RxPDO 1 to 4

| | |
|-------------|----------------------------------|
| Index | 1400h 1401h 1402h 1403h |
| Name | receive PDO parameter |
| Object code | RECORD |
| Data type | PDO CommPar |
| Category | mandatory |

Defined sub-indices

| | | |
|---------------|--|--|
| Subindex | 0 | |
| Name | highest sub-index supported | |
| Data type | UNSIGNED8 | |
| Category | mandatory | |
| Access | R/O | |
| PDO Mapping | not possible | |
| Value Range | 2 | |
| Default Value | 2 | |
| Subindex | 1 | |
| Name | COB-ID used by PDO | |
| Category | mandatory | |
| Access | R/W | |
| PDO Mapping | not possible | |
| Value Range | UNSIGNED32 | |
| Default Value | Index 1400h: 200h + Node-ID Index 1402h: 400h + Node-ID | Index 1401h: 300h + Node-ID Index 1403h: 500h + Node-ID |

Subindex 1 contains the COB-Id of the PDO as a bit coded information:

| Bit | Value | Meaning |
|----------|-------|--|
| 31 | 0 | PDO exists/is valid |
| | 1 | PDO does not exist/is not valid |
| 30 | 0 | RTR allowed on this PDO, not to be used (Can in Automation organisation) |
| | 1 | RTR not allowed on this PDO |
| 29 | 0 | 11 bit-ID (CAN 2.0A) |
| | 1 | 29 bit-ID (CAN 2.0B), not supported |
| 28 to 11 | X | Identifier-bits with 29 bit-ID, not relevant |
| 10 to 0 | X | Bits 10-0 of COB-ID |

| | |
|---------------|-------------------|
| Subindex | 2 |
| Name | transmission type |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED8 |
| Default Value | FFh |

Subindex 2 contains the transmission type of the PDO. There are two ways of setting:

- the value FFh or 255 for event-triggered PDO, which is directly interpreted by reception and taken into actions,
- values from 0 to 240, which cause a SYNC-telegram-controlled interpretation of the PDO contents. Values of 1 to 240 mean, that 0 to 239 SYNC-telegrams are ignored, before one is interpreted. The value 0 means, that only the next SYNC-telegram is interpreted.

19.33.3.21 Objects 1600-1603h: 1st - 4th RxPDO mapping parameter (DS301)

1600h to 1603h for RxPDO 1 to 4.

| | |
|---------------|---|
| Index | 1600h 1601h 1602h 1603h |
| Name | receive PDO mapping |
| Object Code | RECORD |
| Data Type | PDO Mapping |
| Category | mandatory |
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | 0: PDO is not active 1 - 8: PDO activated, mappings are taken only byte-wise |
| Default Value | PDO1: 1 PDO2: 2 PDO3: 2 PDO4: 2 |
| Subindex | 1 - 8 |
| Name | PDO - mapping for the n-th application object |
| Category | Conditional, depends on number and size of object be mapped |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED32 |

| | |
|---------------|-----------|
| Default Value | See below |
|---------------|-----------|

19.33.3.22 Objects 1800-1803h: 1st - 4th TxPDO communication parameter (DS301)

1800h to 1803h for TxPDO 1 to 4.

| | |
|-------------|----------------------------------|
| Index | 1800h 1801h 1802h 1803h |
| Name | transmit PDO parameter |
| Object code | RECORD |
| Data type | PDO CommPar |
| Category | mandatory |

| | |
|---------------|-----------------------------|
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 5 |
| Default Value | 5 |

| | |
|---------------|--|
| Subindex | 1 |
| Name | COB-ID used by PDO |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED32 |
| Default Value | Index 1800h: 180h + Node-ID Index 1801h: 280h + Node-ID Index 1802h: 380h + Node-ID Index 1803h: 480h + Node-ID |

| | |
|---------------|-------------------|
| Subindex | 2 |
| Name | transmission type |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED8 |
| Default Value | FFh |

| | |
|---------------|-----------------------|
| Subindex | 3 |
| Name | inhibit time |
| Category | optional |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED16 (n*1/10ms) |
| Default Value | 0h |

| | |
|---------------|--------------|
| Subindex | 4 |
| Name | reserved |
| Category | optional |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | 0 |
| Default Value | 0 |

| | |
|---------------|-----------------------------|
| Subindex | 5 |
| Name | event timer |
| Category | optional |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED16 (0=not used, ms) |
| Default Value | 0h |

Subindex 1 contains the COB-Id of the PDO as a bit coded information:

| Bit-Number | Value | Meaning |
|------------|-------|--|
| 31 | 0 | PDO exists/is valid |
| | 1 | PDO does not exist/is not valid |
| 30 | 0 | RTR allowed on this PDO, not supported |
| | 1 | RTR not allowed on this PDO, not supported |
| 29 | 0 | 11 bit-ID (CAN 2.0A) |
| | 1 | 29 bit-ID (CAN 2.0B), not supported |
| 28 to 11 | X | Identifier-bits with 29 bit-ID, not relevant |
| 10 to 0 | X | Bits 10-0 of COB-ID |

Subindex 2 contains the transmission type of the PDO. There are two ways of setting:

- A value of FFh or 255d for an event-triggered PDO, which is sent immediately after a change in the mapped application objects. Setting of Subindex 3 or 5 has an influence on the sending of a PDO. With Subindex 3 you can configure, in which minimal time the so configured Transmit-PDOs are sent, if PDO-data contents change (reduction of bus-load). With Subindex 5 (event time) a timer is used, which is reset with every event-triggered sending of this PDO. If there is no change of the PDO-content in this time, the PDO is sent caused by this timer event.
- Values from 0 to 240 cause a SYNC-Telegram controlled sending of the PDO.
- Values from 1 to 240 define how often the SYNC-telegram leads to a sending of a PDO.
- The value 0 means, that only the next SYNC-telegram leads to a sending of the so configured PDOs.

19.33.3.23 Objects 1A00-1A03h: 1st - 4th TxPDO mapping parameter (DS301)

1A00h to 1A03h for TxPDO 1 to 4.

| | |
|---------------|---|
| Index | 1A00h 1A01h 1A02h 1A03h |
| Name | transmit PDO mapping |
| Object Code | RECORD |
| Data Type | PDO Mapping |
| Category | mandatory |
| Subindex | 0 |
| Name | number of mapped application objects in PDO |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | 0: PDO is not active 1 - 8: PDO activated, mappings are taken only byte-wise |
| Default Value | PDO1: 1 PDO2: 2 PDO3: 2 PDO4: 2 |
| Subindex | 1 - 8 |
| Name | PDO - mapping for the n-th application object |
| Category | Conditional, depends on number and size of object be mapped |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED32 |
| Default Value | See below |

19.33.3.24 Object 1C12h: RxPDO assign (DS301)

This object is used to define the mapping for receive direction of EtherCAT data. Either one of the fixed RxPDO mappings 1701h to 1724h is chosen or 1 to 4 of the free mappings 1600h to 1603h.

| | |
|-------------|--------------|
| Index | 1C12h |
| Name | RxPDO assign |
| Object code | ARRAY |
| Data type | UNSIGNED16 |
| Category | optional |

Defined sub-indices

| | |
|---------------|--------------------------------------|
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | 0: pdo not active, 1-4 PDO activated |
| Default Value | 1 |

| | |
|---------------|--------------------|
| Subindex | 1 to 4 |
| Name | Subindex 001..004 |
| Data type | UNSIGNED8 |
| Category | optional |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED16 |
| Default Value | 1701h via ESI-file |

19.33.3.25 Object 1C13h: TxPDO assign (DS301)

This object is used to define the mapping for transmit direction of EtherCAT data. Either one of the fixed TxPDO mappings 1B01h to 1B25h is chosen or 1 to 4 of the free mappings 1A00h to 1A03h.

| | |
|-------------|--------------|
| Index | 1C13h |
| Name | TxPDO assign |
| Object code | ARRAY |
| Data type | UNSIGNED16 |
| Category | optional |

Defined sub-indices

| | |
|---------------|--------------------------------------|
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 0: pdo not active, 1-4 PDO activated |
| Default Value | 1 |

| | |
|---------------|--------------------|
| Subindex | 1 to 4 |
| Name | Subindex 001..004 |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED16 |
| Default Value | 1B01h via ESI-file |

19.33.3.26 Object 2000h: System Warnings

This object is used to show up to three actual warnings with their AKD PDMM- specific warning number.

| | |
|---------------|-------------------------------|
| Index | 2000h |
| Name | System Warnings |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 to 3 |
| Description | DRV.WARNING1 to DRV.WARNINGS3 |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | 0 to 999 |
| Default value | 0 |

19.33.3.27 Object 2001h: System Faults

This object is used to show up to ten actual faults with their AKD PDMM- specific fault number.

| | |
|---------------|-----------------------------|
| Index | 2001h |
| Name | System Faults |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0xA |
| Default value | 0xA |
| Subindex | 1 to A |
| Description | DRV.FAULT1 to DRV.FAULT10 |
| Mode | independent |

| | |
|---------------|--------------|
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | 0 to 999 |
| Default value | 0 |

19.33.3.28 Object 2002h: Manufacturer status bytes

This object delivers the information of the manufacturer status (object 0x1002 sub 0) as four separate, mappable, bytes.

| | |
|---------------|--|
| Index | 2002h |
| Name | Manufacturer status bytes |
| Object code | ARRAY |
| Data type | UNSIGNED8 |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0x4 |
| Default value | 0x4 |
| Subindex | 1 to 4 |
| Description | Manufacturer status byte 1 to Manufacturer status byte 4 |
| Mode | independent |
| Access | R/O |
| PDO mapping | possible |
| Unit | — |
| Value range | 0 to 0xFF |
| Default value | - |

19.33.3.29 Object 2011h: DRV.RUNTIME in seconds

This object delivers the runtime of the drive in seconds.

| | |
|---------------|------------------------|
| Index | 2011h |
| Name | DRV.RUNTIME in seconds |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Access | R/O |
| Unit | — |
| Value range | UNSIGNED32 |
| Default value | 0 |

19.33.3.30 Object 2012h: Fault history: Fault numbers

This object delivers the 20 latest entries of the fault numbers of the fault history table. The latest event can be read via sub-index 1. With new events the list is shifted to higher sub-indices.

| | |
|---------------|---|
| Index | 2012h |
| Name | Fault history: Fault numbers |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0x14h |
| Default value | 0x14h |
| Subindex | 1 to 20 |
| Description | Nth-latest entry in fault number list of fault history table (DRV.FAULHIST) |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | 0 - 999 |
| Default value | 0 |

19.33.3.31 Object 2013h: Fault history: Time stamps

This object delivers the 20 latest entries of the fault time stamps of the fault history table in seconds related to DRV.RUNTIME. The latest event can be read via sub-index 1. With new events the list is shifted to higher sub-indices.

| | |
|---------------|--|
| Index | 2013h |
| Name | Fault history: Time stamps |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0x14h |
| Default value | 0x14h |
| Subindex | 1 to 20 |
| Description | Nth-latest entry in fault time stamp list of fault history table (DRV.FAULTHIST) |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED32 |
| Default value | — |

19.33.3.32 Object 2014-2017h: 1st-4th Mask 1 to 4 for Transmit-PDO

In order to reduce the bus loading with event-triggered PDOs, masking can be used to switch off the monitoring for individual bits in the PDO. In this way it can be arranged, for instance, that actual position values are only signaled once per turn.

This Object masks the PDO-channels 1 to 4. If only two bytes have been defined in a PDO, then it masks just two bytes, although 4 bytes of mask information have been transmitted.

An activated bit in the mask means that monitoring is active for the corresponding bit in the PDO.

| | |
|---------------|----------------------------------|
| Index | 2014h 2015h 2016h 2017h |
| Name | tx_mask 1 to 4 |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 1 |
| Description | tx_mask1 to 4_low |
| Mode | independent |
| Access | R/W |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED32 |
| Default value | FFFFFFFFh |
| Subindex | 2 |
| Description | tx_mask1 to 4_high |
| Mode | independent |
| Access | R/W |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED32 |
| Default value | FFFFFFFFh |

19.33.3.33 Object 2018h: Firmware Version

This object gives all information regarding the firmware version.

Example: Firmware version M_01_00_01_005 would show the numbers 1, 0, 1, 5 in the sub-indices 1 to 4.

| | |
|---------------|------------------|
| Index | 2018h |
| Name | firmware version |
| Object code | ARRAY |
| Data type | UNSIGNED16 |
| Subindex | 1 |
| Description | major version |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED16 |
| Default value | 0 |
| Subindex | 2 |
| Description | minor version |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED16 |
| Default value | 0 |
| Subindex | 3 |
| Description | revision |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED16 |
| Default value | 0 |
| Subindex | 4 |
| Description | branch revision |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED16 |
| Default value | 0 |

19.33.3.34 Object 2026h: ASCII Channel

This object is used to build up an ASCII - communication channel to the drive with 4-byte ASCII-strings.

| | |
|---------------|-----------------------------|
| Index | 2026h |
| Name | ASCII Channel |
| Object code | ARRAY |
| Data type | Visible String |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | 2 |

Subindex 1 is used to send four ASCII-characters to the drive.

| | |
|---------------|----------------|
| Subindex | 1 |
| Description | Command |
| Category | mandatory |
| Access | wo |
| PDO mapping | no |
| Value range | Visible String |
| Default value | — |

Subindex 2 is used to receive four characters from the drive.

| | |
|---------------|----------------|
| Subindex | 2 |
| Description | Response |
| Category | mandatory |
| Access | R/O |
| PDO mapping | no |
| Value range | Visible String |
| Default value | - |

19.33.3.35 Object 204Ch: PV Scaling Factor

This object shall indicate the configured numerator and denominator of the pv scaling factor. The pv scaling factor serves to modify the resolution or directing range of the specified setpoint. It is also included in calculation of the vl velocity demand, and vl velocity actual value. It does not influence the velocity limit function and the ramp function. The value shall have no physical unit and shall be given in the range from -32 768 to +32 767, but the value of 0 shall not be used.

The velocity scaling factor is only active, when bit 4 of FBUS.PARAM05 is set to 1. Otherwise velocities are scaled as 1/1000 rpm.

| | |
|---------------|-------------------------------|
| Index | 204Ch |
| Name | pv scaling factor |
| Object code | ARRAY |
| Data type | INTEGER32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | no |
| Subindex | 1 |
| Description | pv scaling factor numerator |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | +1 |
| Subindex | 2 |
| Description | pv scaling factor denominator |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | +1 |

19.33.3.36 Object 2071h: Target Current

This parameter can be used alternatively to the DS402 parameter 6071h and is the input to the torque controller. The value is scaled in mA (milli Amperes).

| | |
|---------------|--------------------------------------|
| Index | 2071h |
| Name | Target current |
| Object code | VAR |
| Data type | INTEGER 32 |
| Category | optional |
| Access | RW |
| PDO mapping | possible |
| Value range | depends on DRV.IPEAK and MOTOR.IPEAK |
| Default value | 0 |

19.33.3.37 Object 2077h: Current Actual Value

This parameter can be used alternatively to the DS402 parameter 6077h. The value is scaled in mA (milli Amperes).

| | |
|---------------|--------------------------------------|
| Index | 2077h |
| Name | Current actual value |
| Object code | VAR |
| Data type | INTEGER 32 |
| Category | optional |
| Access | RO |
| PDO mapping | possible |
| Value range | depends on DRV.IPEAK and MOTOR.IPEAK |
| Default value | 0 |

19.33.3.38 Object 20A0h: Latch position 1, positive edge

This object is used to output the position or a time, depending on CAP0.MODE, at which the first positive edge occurred on a signal, which can be configured with CAP0.TRIGGER. The latch enable must be active for that purpose(see object 20A4 and 20A5). With CAP0.MODE = 3 the latched position of the encoder index pulse is transferred via this object.

| | |
|---------------|---|
| Index | 20A0h |
| Name | Latch position 1 positive edge CAP0.PLFB, Time capture CAP0.T |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |
| Default value | 0 |

19.33.3.39 Object 20A1h: Latch position 1, negative edge

This object is used to output the position or a time, depending on CAP0.MODE, at which the first negative edge occurred on a signal, which can be configured with CAP0.TRIGGER. The latch enable must be active for that purpose(see object 20A4 and 20A5).

| | |
|---------------|---|
| Index | 20A1h |
| Name | Latch position 1 negative edge CAP0.PLFB, Time capture CAP0.T |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |
| Default value | 0 |

19.33.3.40 Object 20A2h: Latch position 2, positive edge

This object is used to output the position or a time, depending on CAP1.MODE, at which the first positive edge occurred on a signal, which can be configured with CAP1.TRIGGER. The latch enable must be active for that purpose(see object 20A4 and 20A5).

| | |
|---------------|---|
| Index | 20A2h |
| Name | Latch position 2 positive edge CAP1.PLFB, Time capture CAP1.T |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |
| Default value | 0 |

19.33.3.41 Object 20A3h: Latch position 2, negative edge

This object is used to output the position or a time, depending on CAP1.MODE, at which the first negative edge occurred on a signal, which can be configured with CAP1.TRIGGER. The latch enable must be active for that purpose(see object 20A4 and 20A5).

| | |
|---------------|---|
| Index | 20A3h |
| Name | Latch position 2 negative edge CAP1.PLFB, Time capture CAP1.T |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |
| Default value | 0 |

19.33.3.42 Object 20A4h: Latch Control Register

The latch control register is used to enable the latch monitoring of the capture engines 0 and 1. The latch is enabled with a 1 signal and disabled with a 0 signal. Whether or not a latch event has occurred can be recognised by the latch status register (object 20A5).

| | |
|---------------|------------------------|
| Index | 20A4h |
| Name | Latch Control Register |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | rww |
| PDO mapping | possible |
| Value range | 0 to 15 |
| Default value | 0 |

| Bit | Value (bin) | Value (hex) | Description |
|----------|-------------------|-------------|---------------------------------------|
| 0 | 00000000 00000001 | xx01 | Enable extern latch 1 (positive rise) |
| 1 | 00000000 00000010 | xx02 | Enable extern latch 1 (negative rise) |
| 2 | 00000000 00000100 | xx04 | Enable extern latch 2 (positive rise) |
| 3 | 00000000 00001000 | xx08 | Enable extern latch 2 (negative rise) |
| 4 to 7 | | | Reserve |
| 8 | 00000001 00000000 | 01xx | Read external latch 1 (positive rise) |
| 9 | 00000010 00000000 | 02xx | Read external latch 1 (negative rise) |
| 10 | 00000011 00000000 | 03xx | Read external latch 2 (positive rise) |
| 11 | 00000100 00000000 | 04xx | Read external latch 2 (negative rise) |
| 12 to 15 | | | Reserve |

19.33.3.43 Object 20A5h: Latch Status Register

The latch status register is used to look for the states of the capture engines 0 and 1.

| | |
|---------------|-----------------------|
| Index | 20A5h |
| Name | Latch Status Register |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | rwr |
| PDO mapping | possible |
| Value range | - |
| Default value | 0 |

| Bit | Value (bin) | Value (hex) | Description |
|--------|----------------------|-------------|--|
| 0 | 00000000 00000001 | zz01 | External latch 1 valid (positive rise) |
| 1 | 00000000 00000010 | zz02 | External latch 1 valid (negative rise) |
| 2 | 00000000 00000100 | zz04 | External latch 2 valid (positive rise) |
| 3 | 00000000 00001000 | zz08 | External latch 2 valid (negative rise) |
| 4 to 7 | | | Reserve |
| 8 | 00000001 00000000 | z1zz | Acknowledge value external latch 1 (positive rise) |
| 9 | 00000010 00000000 | z2zz | Acknowledge value external latch 1 (negative rise) |
| 10 | 00000011 00000000 | z3zz | Acknowledge value external latch 2 (positive rise) |
| 11 | 00000100 00000000 | z4zz | Acknowledge value external latch 2 (negative rise) |
| 12 | 00010000 00000000 | 1zzz | State Digital Input 4 |
| 13 | 00100000 00000000 | 2zzz | State Digital Input 3 |
| 14 | 01000000 00000000 | 4zzz | State Digital Input 2 |
| 15 | 10000000 00000000 | 8zzz | State Digital Input 1 |

19.33.3.44 Object 20A6h: Latch position 1, positive or negative edge

This object is used to output the position or a time, depending on CAP0.MODE, at which the first positive or negative edge occurred on a signal, that can be configured with CAP0.TRIGGER. Latch enable must be active for that purpose (see object 20A4 and 20A5).

| | |
|---------------|---|
| Index | 20A6h |
| Name | Latch position 1 positive or negative CAP0.PLFB |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | ro |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |
| Default value | 0 |

19.33.3.45 Object 20A7h: Latch position 2, positive or negative edge

This object is used to output the position or a time, depending on CAP1.MODE, at which the first positive or negative edge occurred on a signal, that can be configured with CAP1.TRIGGER. Latch enable must be active for that purpose (see object 20A4 and 20A5).

| | |
|---------------|---|
| Index | 20A7h |
| Name | Latch position 2 positive or negative CAP1.PLFB |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | ro |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |
| Default value | 0 |

19.33.3.46 Object 20B8h: Reset of changed input information

This object is used in PDOs to reset the state change information for the digital inputs shown in the Bits 24 to 30 in the object 60FD. Bit 0 to 6 are used to reset the information of the digital input 1 to 7.

| | |
|---------------|------------------------------------|
| Index | 20B8h |
| Name | Reset of changed input information |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | rw |
| PDO mapping | possible |
| Value range | UNSIGNED16 |
| Default value | 0 |

19.33.3.47 Object 345Ah: Brake Control

These objects implement the possibility to control the brake directly, overriding the drive logic. When the brake state is controlled by the fieldbus, the drive state (enabled, disabled, faulted) will have no effect on the brake - the fieldbus will be in control.

NOTICE

Applying or releasing the brake at the wrong time can be a safety hazard and can destroy your mechanic as well as drive or motor. Unexpected behaviour might be possible. It is the responsibility of the customer using this mode to use this function appropriately.

When fieldbus control is disabled, the drive will control the brake as defined by existing AKD PDMM brake related parameters. As soon as fieldbus control is enabled, the Brake Command received over the field bus will take effect. So, if the Brake Command is set to APPLY and the current state is RELEASE, the brake will begin to apply .

The default value of the fieldbus control will be disabled, so that the drive is always in control until the fieldbus is operational. It is recommended that this bit remain 0 except for special operating conditions where the fieldbus will control the brake. When fieldbus communication is lost, the drive will regain control of the brake if the fieldbus had previously taken control.

| Enable Fieldbus Control | Serious Failure condition present | Brake Command | Fieldbus Control Status | Controlled by... | Final Brake State |
|-------------------------|-----------------------------------|---------------|-------------------------|------------------|-------------------|
| 0 | x | x | 0 | Drive | Drive |
| 1* | no | 0 | 1 | Fieldbus | Applied |
| 1* | no | 1 | 1 | Fieldbus | Released |
| x | yes | any | 0 | Drive | Drive |

1* indicates that a rising edge was seen since the last time the drive applied the brake

| | |
|-------------|---------------|
| Index | 345Ah |
| Name | Brake Control |
| Object code | ARRAY |
| Data type | UNSIGNED16 |
| Category | optional |

Defined sub-indices

| | |
|---------------|-----------------------------|
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 2 |
| Default Value | 2 |

| | |
|---------------|-----------------------|
| Subindex | 1 |
| Name | Brake Control Command |
| Category | optional |
| Access | R/W |
| PDO Mapping | possible |
| Value Range | UNSIGNED16 |
| Default Value | 0 |

With subindex 1 the brake is controlled. Bit definition:

| Bit | Name | Description |
|-----|-------------------------|--|
| 0 | Enable fieldbus control | 0 - brake is not controlled via this object 1 - enable fieldbus control via this object. This function works edge triggered, i.e. this bit has to have a 0 -> 1 transition to activate the brake control functionality. After a fault the functionality is reset and has to be activated again. The activation can be controlled by subindex 2 bit 0. |
| 1 | Brake Command | This command bit is only active, if the functionality was activated via bit 0. The function is as follows: 0 - apply the brake 1 - release the brake |

| | |
|---------------|-----------------------|
| Subindex | 2 |
| Name | Brake Status Response |
| Category | optional |
| Access | R/O |
| PDO Mapping | possible |
| Value Range | UNSIGNED16 |
| Default Value | 0 |

With subindex 2 the brake status can be checked. Bit definition:

| Bit | Name | Description |
|-----|-------------------------|---|
| 0 | Fieldbus control Status | 0 - brake control via 0x345A is disabled or not possible due to drive failure. 1 - enable fieldbus control via this object. This function works edge triggered, i.e. this bit has to have a 0 -> 1 transition to activate the brake control functionality. After a fault the functionality is reset and has to be activated again. The activation can be controlled by subindex 2 bit 0. |
| 1 | Brake Status | 0 - apply the brake 1 - release the brake Note: When the brake is applied or released, there is a time delay MOTOR.TBRAKEAPP or MOTOR.TBRAKEREL, after the receipt of the command before this status bit changes. The status is always reported: it is not affected by fieldbus control. |

| | | |
|---|------------------|---|
| 2 | STO Status | 0 - STO is not active (drive may be enabled) 1 - STO is active (drive can not be enabled) |
| 3 | HW Enable Status | 0 - HW enable is disabled, drive function can not be enabled 1 - HW enable is enabled, drive function can be enabled |

19.33.3.48 Object 3474h: Parameters for digital inputs

This set of objects is used to set extended parameters for some digital input functions. The parameters can be used for different DINx.MODEs. Therefore the scaling might be different or no scaling is used at all.

Two subindices build an access object to one of these parameters, because they are 64-bit numbers internally, e.g. object 3474 sub 1 gives access to the low 32 bits of DIN1.PARAM whereas 3474 sub 8 sub 4 gives access to the high 32 bits.

If access to the whole 64 bit number is needed the higher bits must be written first. The access to the lower 32 bits then writes the parameter. If the to be written value fits into 32 bit, only the lower part needs to be written. The most-significant bit is then taken as sign-bit for the number.

| | |
|---------------|--|
| Index | 3474h |
| Name | DINx.PARAM |
| Object code | Array |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0xE |
| Default value | 0xE |
| Subindex | 1 to 7 1 to 3 |
| Description | DINx.PARAM low 32 bits, x = 1 .. 7 1 .. 3 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0 |
| Subindex | 8 to 0xE 4 to 6 |
| Description | DINx.PARAM high 32 bits, x = 1 .. 7 1 .. 3 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0 |

19.33.3.49 Object 3475h: Parameters for digital outputs

This set of objects is used to set extended parameters for some digital output functions. The parameters can be used for different DOUTx.MODEs. Therefore the scaling might be different or no scaling is used at all.

Two subindices build an access object to one of these parameters, because they are 64-bit numbers internally, e.g. object 3475 sub 1 gives access to the low 32 bits of DOUT1.PARAM whereas 3475 sub 32 gives access to the high 32 bits.

If access to the whole 64 bit number is needed the higher bits must be written first. The access to the lower 32 bits then writes the parameter. If the to be written value fits into 32 bit, only the lower part needs to be written. The most-significant bit is then taken as sign-bit for the number.

| | |
|---------------|--------------------------------------|
| Index | 3475h |
| Name | DOUTx.PARAM |
| Object code | Array |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0x4 |
| Default value | 0x4 |
| Subindex | 1 to 2 |
| Description | DOUTx.PARAM low 32 bits, x = 1 .. 2 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0 |
| Subindex | 3 to 42 |
| Description | DOUTx.PARAM high 32 bits, x = 1 .. 2 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0 |

19.33.3.50 Object 3496h: Fieldbus synchronization parameters

This set of objects is used to set or read parameters for the fieldbus synchronization used in the interpolated position mode (7) and the cyclic-modes 8 etc. The synchronization between a fieldbus master and the AKD PDMM is similar in all the supported fieldbus systems.

The AKD PDMM internal 16[kHz] interrupt function is responsible for calling the PLL function. This PLL function is called once per fieldbus cycle (set by object 60C2 sub 1 and 2). If the fieldbus sample period is for example 1[ms], the PLL code is called every 16th time of the 16[kHz] IRQ of the AKD PDMM.

Once in a fieldbus sample the SYNC-telegram must arrive, which resets a PLL counter in the Drive. After some time the already mentioned PLL function is called and reads back the time from that PLL counter.

Depending on the measured time the PLL function extends (in case that the measured time is too low) or lowers (in case that the measured time is too high) the sample time of the upcoming 16[kHz] tasks for one fieldbus sample by a selectable value (object 3496 sub 4) in order to move the PLL function closer to the expected distance (object 3496 sub 1).

Beside the objects mentioned here the parameter FBUS.SAMPLEPERIOD is important, which is set by object 60C2 sub 1 and 2. This setting is required in order to share the fieldbus sample time with the slave. This information is e.g. needed for being able to call the AKD PDMM internal PLL function once per fieldbus sample.

| | |
|---------------|---------------------------------|
| Index | 3496h |
| Name | FBUS synchronization parameters |
| Object code | Array |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0x4 |
| Default value | 0x4 |
| Subindex | 1 |
| Description | FBUS.SYNCDIST |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 320000 [ns] |

Sub 1 is the expected time distance in nano seconds between clearing the PLL counter and calling the PLL function.

| | |
|---------------|--------------|
| Subindex | 2 |
| Description | FBUS.SYNCACT |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 320000 [ns] |

Sub 2 is the actual time distance in nano seconds between clearing the PLL counter and calling the PLL function.

| | |
|---------------|--------------|
| Subindex | 3 |
| Description | FBUS.SYNCWND |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 70000 [ns] |

Sub 3 is a window, which is used in order to consider the drive as being synchronized. The AKD PDMM is considered as synchronized in the following case:

$$\text{FBUS.SYNCDIST} - \text{FBUS.SYNCWND} < \text{FBUS.SYNCACT} < \text{FBUS.SYNCDIST} + \text{FBUS.SYNCWND}$$

| | |
|---------------|---------------|
| Subindex | 4 |
| Description | FBUS.COMPTIME |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 150 [ns] |

Sub 4 value indicates the time, which is used for extending or lowering the sample rate of the AKD PDMM internal 16[kHz] IRQ, which is responsible for calling the PLL function. The default sample time is $32 * 1/16$ [kHz] = 2[ms].

The sample time of the AKD PDMM high prior interrupt is determined by $62.5[\mu\text{s}] - \text{FBUS.COMPTIME}$ if $\text{FBUS.SYNCACT} > \text{FBUS.SYNCDIST}$.

The sample time of the AKD PDMM high prior interrupt is determined by $62.5[\mu\text{s}] + \text{FBUS.COMPTIME}$ if $\text{FBUS.SYNCACT} < \text{FBUS.SYNCDIST}$.

19.33.3.51 Object 6040h: Control word (DS402)

The control commands are built up from the logical combination of the bits in the control word and external signals (e.g enable output stage). The definitions of the bits are shown below:

| | |
|---------------|--------------|
| Index | 6040h |
| Name | control word |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Access | R/W |
| PDO mapping | possible |
| Unit | — |
| Value range | 0 to 65535 |
| EEPROM | no |
| Default value | 0 |

Bit assignment im control word

| Bit | Name | Bit | Name |
|-----|---|-----|-----------------------|
| 0 | Switch on | 8 | Pause/halt |
| 1 | Disable Voltage | 9 | reserved |
| 2 | Quick Stop | 10 | reserved |
| 3 | Enable Operation | 11 | reserved |
| 4 | Operation mode specific | 12 | reserved |
| 5 | Operation mode specific | 13 | Manufacturer-specific |
| 6 | Operation mode specific | 14 | Manufacturer-specific |
| 7 | Reset Fault (only effective for faults) | 15 | Manufacturer-specific |

Commands in the control word

| Command | Bit 7 Fault Reset | Bit 3 Enable Operation | Bit 2 Quick Stop | Bit 1 Disable Voltage | Bit 0 Switch on | Transitions |
|------------------------|-------------------------|------------------------------|------------------------|-----------------------------|-----------------------|--------------|
| Shutdown | X | X | 1 | 1 | 0 | 2, 6, 8 |
| Switch on | X | X | 1 | 1 | 1 | 3 |
| Disable Voltage | X | X | X | 0 | X | 7, 9, 10, 12 |
| Quick Stop | X | X | 0 | 1 | X | 7, 10, 11 |
| Disable Oper- ation | X | 0 | 1 | 1 | 1 | 5 |
| Enable Operation | X | 1 | 1 | 1 | 1 | 4, 16 |
| Fault Reset | 1 | X | X | X | X | 15 |

Bits marked by an X are irrelevant.

Mode-dependent bits in the control word

The following table shows the mode-dependent bits in the control word. Only manufacturer-specific modes are supported at present. The individual modes are set by Object 6060_h Modes of operation.

| Operation mode | No. | Bit 4 | Bit 5 | Bit 6 |
|---------------------------------|-----|------------------------|------------------------|-------------------|
| Profile Position Mode (pp) | 01h | new_setpoint | change_set_immediately | absolute/relative |
| Profile Velocity Mode (pv) | 03h | reserved | reserved | reserved |
| Profile Torque Mode (tq) | 04h | reserved | reserved | reserved |
| Homing Mode (hm) | 06h | homing_operation_start | reserved | reserved |
| Interpolated Position Mode (ip) | 07h | Enable Interpolation | reserved | reserved |
| Cyclic sync position Mode (csp) | 08h | reserved | reserved | reserved |

Description of the remaining bits in the control word

The remaining bits in the control word are described below.

Bit 8 Pause If Bit 8 is set, then the drive halts (pauses) in all modes. The setpoints (speed for homing or jogging, motion task number, setpoints for digital mode) for the individual modes are retained.

Bit 9,10 These bits are reserved for the drive profile (DS402).

Bit 13, 14, 15 These bits are manufacturer-specific, and reserved at present.

19.33.3.52 Object 6041h: Status word (DS402)

The momentary state of the state machine can be read out with the aid of the status word.

| | |
|---------------|-------------|
| Index | 6041h |
| Name | Status word |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Access | R/W |
| PDO mapping | possible |
| Unit | — |
| Value range | 0 to 65535 |
| EEPROM | yes |
| Default value | 0 |

Bit assignment in the status word

| Bit | Name | Bit | Name |
|-----|--------------------|-----|------------------------------------|
| 0 | Ready to switch on | 8 | STO – Safe Torque Off |
| 1 | Switched on | 9 | Remote |
| 2 | Operation enabled | 10 | Target reached |
| 3 | Fault | 11 | Internal limit active |
| 4 | Voltage enabled | 12 | Operation mode specific (reserved) |
| 5 | Quick stop | 13 | Operation mode specific (reserved) |

| | | | |
|---|--------------------|----|----------------------------------|
| 6 | Switch on disabled | 14 | Manufacturer-specific (reserved) |
| 7 | Warning | 15 | Manufacturer-specific (reserved) |

States of the state machine

| State | Bit 6 switch on disabled | Bit 5 quick stop | Bit 3 fault | Bit 2 operation enabled | Bit 1 switched on | Bit 0 ready to switch on |
|------------------------|--------------------------------|---------------------|----------------|-------------------------------|-------------------------|-----------------------------------|
| Not ready to switch on | 0 | X | 0 | 0 | 0 | 0 |
| Switch on disabled | 1 | X | 0 | 0 | 0 | 0 |
| Ready to switch on | 0 | 1 | 0 | 0 | 0 | 1 |
| Switched on | 0 | 1 | 0 | 0 | 1 | 1 |
| Operation enabled | 0 | 1 | 0 | 1 | 1 | 1 |
| Fault | 0 | X | 1 | 0 | 0 | 0 |
| Fault reaction active | 0 | X | 1 | 1 | 1 | 1 |
| Quick stop active | 0 | 0 | 0 | 1 | 1 | 1 |

Bits marked by X are irrelevant

Description of the remaining bits in the status word

Bit 4: voltage_enabled The DC-link voltage is present if this bit is set.

Bit 7: warning There are several possible reasons for Bit 7 being set and this warning being produced. The reason of a warning can be seen by the Error code of the Emergency message, which is sent on the bus caused by this warning.

Bit 9: The remote-bit is set by the telnet command FBUS.REMOTE. The default state is 1 indicating that the power stage shall be only controlled by the DS402 control word. For special actions via telnet like tuning or commutation finding, FBUS.REMOTE shall be set to 0 via telnet to inform the fieldbus master.

Bit 10: target_reached This is set when the drive has reached the target position. In profile position mode the position window is set via MT.TPOSWND, in homing mode via HOME.TPOSWND.

Bit 11: internal_limit_active This bit specifies that a movement was or is limited. In different modes, different warnings cause the bit to be set. The following assignments exist:

| Mode of operation | Warnings which set Bit 11 |
|-------------------|------------------------------|
| all | n04, n06, n07, n10, n11, n14 |
| 0x1 (PP), 0x88 | n03, n08, n09, n20 |

Mode-dependent bits in the status word

The following table shows the mode-dependent bits in the status word. The individual modes are set by " Object 6060h: Modes of Operation (DS402)" (→ p. 1053).

| Operation mode | No. | Bit 12 | Bit 13 |
|---------------------------------|-----|--|-----------------|
| Profile Position Mode (pp) | 01h | setpoint acknowledge | following error |
| Homing Mode (hm) | 06h | homing attained | homing error |
| Interpolated Position Mode (ip) | 07h | ip mode active | following error |
| Cyclic sync position Mode (csp) | 08h | This bit stays on 1 as long as the drive is following the position set-points. | following error |

19.33.3.53 Object 605Ah: Quick stop option code (DS402)

This object defines the action, which is taken as quick stop function.

| | |
|---------------|------------------------|
| Index | 605Ah |
| Name | Quick stop option code |
| Object code | VAR |
| Data type | INTEGER16 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | 1, 2, 5, 6 |
| Default value | 2 |

Supported codes:

| Bit | Description |
|-----|--|
| 1 | Slow down on slow down ramp and transit into Switch On Disabled |
| 2 | Slow down on quick stop ramp and transit into Switch On Disabled |
| 5 | Slow down on slow down ramp and stay in Quick Stop Active |
| 6 | Slow down on quick stop ramp and stay in Quick Stop Active |

19.33.3.54 Object 6060h: Modes of Operation (DS402)

This object is used to set the mode, which can be read out by Object 6061h. Two types of operating mode are used:

- manufacturer-specific operating modes
- operating modes as per CANopen drive profile DS402

These operating modes are defined in the CANopen drive profile DS402. After the mode has been changed, the corresponding setpoint must be set once more (for instance, the homing velocity in the mode homing_setpoint). If the position or jogging mode is stored, then the Homing mode is set after a RESET of the drive.

NOTE

An operating mode only becomes valid when it can be read by Object 6061h.

**! WARNING****Automatic Start**

Risk of death or serious injury for humans working in the machine. The drive could move unexpectedly. When the drive is enabled, a mode change is only permissible at zero speed.

- Never change the mode while the motor is running!
- Set the speed setpoint to 0 before changing over.

| | |
|---------------|------------------------------|
| Index | 6060h |
| Name | mode of operation |
| Object code | VAR |
| Data type | INTEGER8 |
| Category | mandatory |
| Access | R/W |
| PDO mapping | possible |
| Value range | -3, -2, -1, 1, 3, 4, 6, 7, 8 |
| Default value | — |

Supported modes (negative values are manufacturer specific modes):

| Value (hex) | Mode |
|-------------|----------------------------------|
| -3 | Electronic gearing mode |
| -2 | Analog velocity mode |
| -1 | Analog current mode |
| 1 | Profile position mode |
| 3 | Profile velocity mode |
| 4 | Profile torque mode |
| 6 | Homing mode |
| 7 | Interpolated position mode |
| 8 | Cyclic synchronous position mode |

19.33.3.55 Object 6061h: Modes of Operation Display (DS402)

This object can be used to read the mode that is set by Object 6060h. An operating mode only becomes valid when it can be read by Object 6061h (see also Object 6060h).

| | |
|---------------|------------------------------|
| Index | 6061h |
| Name | mode of operation display |
| Object code | VAR |
| Data type | INTEGER8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | possible |
| Value range | -3, -2, -1, 1, 3, 4, 6, 7, 8 |
| Default value | — |

19.33.3.56 Object 6063h: position actual value* (DS402)

The object position actual value provides the momentary actual position in increments. The resolution is defined with Object 608F as power-of-two number.

| | |
|---------------|---|
| Index | 6063h |
| Name | position actual value |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | pc, pp |
| Access | R/W |
| PDO mapping | possible |
| Unit | increments (1 turn = $2^{FB1.PSCALE}$) |
| Value range | (-2^{31}) to $(2^{31}-1)$ |
| Default value | 2^{20} |
| EEPROM | no |

19.33.3.57 Object 6064h: position actual value (DS402)

The object position actual value provides the actual position. The resolution can be altered by the gearing factors of the position controller (Object 6091/6092).

| | |
|-------------|------------------------------|
| Index | 6064h |
| Name | position actual value, PL.FB |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | pp, csp |
| Access | R/W |
| PDO mapping | possible |
| Unit | position units |

| | |
|---------------|-----------------------------|
| Value range | (-2^{31}) to $(2^{31}-1)$ |
| Default value | — |
| EEPROM | no |

19.33.3.58 Object 6065h: Following error window

The following error window defines a range of tolerated position values symmetrically to the position demand value. A following error might occur when a drive is blocked, unreachable profile velocity occurs, or at wrong closed loop coefficients. If the value of the following error window is 0, the following control is switched off.

| | |
|---------------|------------------------|
| Index | 6065h |
| Name | Following error window |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0 |

19.33.3.59 Object 606Ch: Velocity actual value (DS402)

The object velocity actual value represents the actual speed.

| | |
|---------------|---|
| Index | 606Ch |
| Name | velocity actual value, VL.FB |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | pv |
| Access | R/O |
| PDO mapping | possible |
| Unit | velocity units (SDO is in user units and the PDO is in RPM) |
| Value range | (-2^{31}) to $(2^{31}-1)$ |
| Default value | — |
| Float scaling | 1000:1 |
| EEPROM | no |

19.33.3.60 Object 6071h: Target torque (DS402)

This parameter is the input value for the torque controller in profile torque mode and the value is given per thousand (1‰) of rated torque.

| | |
|---------------|---|
| Index | 6071h |
| Name | Target torque |
| Object code | VAR |
| Data type | INTEGER16 |
| Category | conditional; mandatory, if tq supported |
| Access | R/W |
| PDO mapping | possible |
| Value range | INTEGER16 |
| Default value | 0 |

19.33.3.61 Object 6073h: Max current (DS402)

This value represents the maximum permissible torque creating current in the motor and is given per thousand (1‰) of rated current.

| | |
|---------------|--------------|
| Index | 6073h |
| Name | Max current |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED16 |
| Default value | 0 |

19.33.3.62 Object 6077h: Torque actual value (DS402)

The torque actual value corresponds to the instantaneous torque in the drive motor. The value is given per thousand (1‰) of rated torque. This object reflects the actual current (DRV.ICONT).

| | |
|---------------|---------------------|
| Index | 6077h |
| Name | Torque actual value |
| Object code | VAR |
| Data type | INTEGER16 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER16 |
| Default value | 0 |

19.33.3.63 Object 607Ah: Target position (DS402)

The object target position defines the target position for the drive. The target position is interpreted as a relative distance or an absolute position, depending on Bit 6 of the control word. The type of relative movement can be further defined by the manufacturer-specific parameter 35B9h Subindex 0. Other properties like following motion tasks can be set with this object as well. The mechanical resolution is set via the scaling objects 6091h and 6092h.

| | |
|---------------|-------------------------------|
| Index | 607Ah |
| Name | target position, MT.P |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | pp, csp |
| Access | R/W |
| PDO mapping | possible |
| Unit | user-defined |
| Value range | $-(2^{31}-1)$ to $(2^{31}-1)$ |
| Default value | — |

19.33.3.64 Object 607Ch: Homing offset (DS402)

The reference offset (home offset) is the difference between the zero position for the application and the zero point of the machine. All subsequent absolute motion tasks take account of the reference offset.

| | |
|---------------|-----------------------------|
| Index | 607Ch |
| Name | home offset, HOME.P |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | hm |
| Access | R/W |
| PDO mapping | not possible |
| Unit | user-defined |
| Value range | (-2^{31}) to $(2^{31}-1)$ |
| Default value | 0 |

19.33.3.65 Object 607Dh: Software position limit (DS402)

Software position limit contains the sub-parameters min position limit and max position limit. New target positions are checked against these limits. The limits are relative to the machine home position, which is the result of homing (including the home offset (Object 607Ch)). As default the software position limits are switched off. Changed values must be saved and the drive must be restarted to take enable the new the software limits.

| | |
|---------------|--------------------------------------|
| Index | 607Dh |
| Name | Software position limit, SWLS.LIMIT0 |
| Object code | ARRAY |
| Data type | INTEGER32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | 2 |
| Subindex | 1 |
| Description | min position limit 1, SWLS.LIMIT0 |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | 0 (switched off) |
| Subindex | 2 |
| Description | Min Position Limit 2, SWLS.LIMIT1 |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | 0 (switched off) |

19.33.3.66 Object 6081h: Profile velocity (DS402)

The profile velocity is the final velocity that should be reached after the acceleration phase of a motion task.

| | |
|---------------|------------------------|
| Index | 6081h |
| Name | profile velocity, MT.V |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Mode | pp |
| Access | R/W |
| PDO mapping | possible |
| Unit | speed units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | 10 |

19.33.3.67 Object 6083h: Profile acceleration (DS402)

The acceleration ramp (profile acceleration) is given in units that are defined by the user (position units per s²). The position units are scaled via the objects 6091 and 6092. This object is connected to the AKD PDMM-parameter DRV.ACC in the Profile Velocity Mode and to the motion task parameter MT.ACC in all other modes.

| | |
|---------------|---|
| Index | 6083h |
| Name | profile acceleration, MT.ACC (DRV.ACC in Profile Velocity Mode) |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Mode | pp, pv |
| Access | R/W |
| PDO mapping | possible |
| Unit | acceleration units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | 0 |

19.33.3.68 Object 6084h: Profile deceleration (DS402)

The braking/deceleration ramp is handled in the same way as the acceleration ramp (" Object 6083h: Profile acceleration (DS402)" (→ p. 1059)).

| | |
|-------------|---|
| Index | 6084h |
| Name | profile deceleration, MT.DEC (DRV.DEC in Profile Velocity Mode) |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Mode | pp, pv |
| Access | R/W |
| PDO mapping | possible |

| | |
|---------------|--------------------|
| Unit | deceleration units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | 0 |

19.33.3.69 Object 6087h Torque slope (DS402)

This object defines the rate of change of torque. The value is given in units of per thousand of rated torque per second. The minimum rate settable for the AKD PDMM is equivalent to the value of DRV.ICONT (= 1000 per mille) per ~ 420 milliseconds equivalent to a value of ~ 2385 per mille / second.

| | |
|---------------|-------------------------------|
| Index | 6087h |
| Name | Torque slope |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | mandatory, if tq is supported |
| Access | R/W |
| PDO mapping | possible |
| Value range | UNSIGNED32 |
| Default value | - |

19.33.3.70 Object 608Fh: Position encoder resolution (DS402)

The position encoder resolution defines the ratio of encoder increments per motor revolution on the CANopen end. Encoder increments are set either directly by subindex 1 (only powers of 2 available) or implicit by writing to the parameter FB1.PSCALE.

| | |
|---------------|-----------------------------|
| Index | 608Fh |
| Name | Position encoder resolution |
| Object Code | ARRAY |
| Data Type | UNSIGNED 32 |
| Category | optional |
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 2 |
| Default Value | 2 |
| Subindex | 1 |
| Name | Encoder increments |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 2 ²⁰ |
| Subindex | 2 |

| | |
|---------------|-------------------|
| Name | Motor revolutions |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 1 |

19.33.3.71 Object 6091h: Gear Ratio (DS402)

The gear ratio defines the ratio of feed in position units per driving shaft revolutions. This includes the gear if present.

gear ratio = motor shaft revolutions / driving shaft revolutions

| | |
|-------------|-------------|
| Index | 6091h |
| Name | Gear Ratio |
| Object Code | ARRAY |
| Data Type | UNSIGNED 32 |
| Category | optional |

| | |
|---------------|-----------------------------|
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 2 |
| Default Value | 2 |

| | |
|---------------|------------------|
| Subindex | 1 |
| Name | Motor revolution |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 1 |

| | |
|---------------|-------------------|
| Subindex | 2 |
| Name | Shaft revolutions |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 1 |

19.33.3.72 Object 6092h: Feed constant (DS402)

The feed constant defines the ratio of feed in position units per driving shaft revolutions. This includes the gear if present.

| | |
|-------------|---------------|
| Index | 6092h |
| Name | Feed constant |
| Object Code | ARRAY |
| Data Type | UNSIGNED 32 |
| Category | optional |

| | |
|---------------|-----------------------------|
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 2 |
| Default Value | 2 |

| | |
|---------------|--------------|
| Subindex | 1 |
| Name | Feed |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 1 |

| | |
|---------------|-------------------|
| Subindex | 2 |
| Name | Shaft revolutions |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 1 |

19.33.3.73 Object 6098h: Homing method (DS402)

| | |
|---------------|------------------------------------|
| Index | 6098h |
| Name | homing method, HOME.MODE, HOME.DIR |
| Object code | VAR |
| Data type | INTEGER8 |
| Mode | hm |
| Access | R/W |
| PDO mapping | not possible |
| Unit | position units |
| Value range | -128 to 127 |
| Default value | 0 |

Description of the homing methods

Choosing a homing method by writing a value to homing method (Object 6098h) will clearly establish:

- the homing signal (P-Stop, N-Stop, reference switch)
- the direction of actuation

and where appropriate

- the position of the index pulse.

The reference position is give by the reference offset (Object 607Ch).

A detailed description of the types of homing movement can be found in the description of [EWV¹](#).

The following homing methods are supported:

| Method as per DS402 | Brief description: Homing | command |
|---------------------|---|---------------------------------------|
| -128 to -8 | reserved | — |
| -7 | find home input in negative direction, then find next feedback zero | HOME.MODE=18, HOME.DIR=0 |
| -6 | find home input in positive direction, then find next feedback zero | HOME.MODE=18, HOME.DIR = 1 |
| -5 | set reference point at actual position and store value in NVRAM | HOME.MODE=17, HOME.DIR not changed |
| -4 | find reference switch with fast velocity (6099h sub1) and home on reference switch with low velocity (6099h sub 2), positive count direction | HOME.MODE=16, HOME.DIR=0 |
| -3 | find reference switch with fast velocity (6099h sub 1) and home on reference switch with low velocity (6099h sub 2), negative count direction | HOME.MODE=16, HOME.DIR=0 |
| -2 to -1 | reserved | — |
| 0 | reserved | — |
| 1 | homing to negative limit switch, with zeroing, negative count direction | HOME.MODE=2, HOME.DIR=0 |

¹Embedded Workbench Views

| Method as per DS402 | Brief description: Homing | command |
|---------------------|---|------------------------------|
| 2 | homing to positive limit switch, with zeroing, positive count direction | HOME.MODE=2, HOME.DIR=1 |
| 3 to 7 | not supported | — |
| 8 | homing to reference switch, with zeroing, positive count direction | HOME.MODE=5, HOME.DIR=1 |
| 9 to 11 | not supported | — |
| 12 | homing to reference switch, with zeroing, negative count direction | HOME.MODE=5, HOME.DIR=0 |
| 13 to 14 | not supported | — |
| 15 to 16 | reserved | — |
| 17 | homing to negative limit switch, without zeroing, negative count direction | HOME.MODE=1, HOME.DIR=0 |
| 18 | homing to negative limit switch, without zeroing, positive count direction | HOME.MODE=1, HOME.DIR=1 |
| 19 to 23 | not supported | — |
| 24 | homing to reference switch, without zeroing, positive count direction | HOME.MODE=4, HOME.DIR=1 |
| 25 to 27 | not supported | — |
| 28 | homing to reference switch, without zeroing, negative count direction | HOME.MODE=4, HOME.DIR=0 |
| 29 to 30 | not supported | — |
| 31 to 32 | reserved | — |
| 33 | homing within a single turn, negative count direction. If the feedback has an index pulse, HOME.MODE 11 will be used. | HOME.MODE=7,11 HOME.DIR=0 |
| 34 | homing within a single turn, positive count direction. If the feedback has an index pulse, HOME.MODE 11 will be used. | HOME.MODE=7,11 HOME.DIR=1 |
| 35 | set reference point at present position | HOME.MODE=0, HOME.DIR=0 |
| 36 to 127 | reserved | — |

19.33.3.74 Object 6099h: Homing speeds (DS402)

| | |
|-------------|--|
| Index | 6099h |
| Name | homing speeds |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 1 |
| Description | speed during search for switch, HOME.V |
| Mode | hm |
| Access | R/W |

| | |
|---------------|---|
| PDO mapping | not possible |
| Unit | velocity units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | equivalent 60 rpm |
| Subindex | 2 |
| Description | speed during search for zero, HOME.FEEDRATE |
| Mode | hm |
| Access | R/W |
| PDO mapping | not possible |
| Unit | velocity units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | 50% of Object 6099 sub 1 |

19.33.3.75 Object 609Ah: Homing acceleration (DS402)

| | |
|---------------|---------------------|
| Index | 609Ah |
| Name | homing acceleration |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Mode | hm |
| Access | R/W |
| PDO mapping | not possible |
| Unit | acceleration units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | 0 |

19.33.3.76 Object 60B1h: Velocity Offset

This object provides the offset of the velocity value in cyclic synchronous position mode. It is scaled via the object 204Ch.

| | |
|---------------|-----------------|
| Index | 60B1h |
| Name | Velocity Offset |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/W |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Default value | 0 |

19.33.3.77 Object 60B2h: Torque Offset

This object (IL.BUSFF) provides the offset of the commanded torque from a bus network connected to the drive. Scaling is 1/1000 of rated torque.

| | |
|---------------|---------------|
| Index | 60B2h |
| Name | Torque Offset |
| Object code | VAR |
| Data type | INTEGER16 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER16 |
| Default value | 0 |

19.33.3.78 Object 60B8h: Touch probe function

This object indicates the configured function of the touch probe.

| | |
|---------------|----------------------|
| Index | 60B8h |
| Name | Touch probe function |
| Object code | Variable |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | R/W |
| PDO Mapping | yes |
| Value range | UNSIGNED16 |
| Default value | 0 |

Definition of the possible functions:

| Bit | Value | Meaning |
|--------|-------|--|
| 0 | 0 | Switch off touch probe 1 |
| | 1 | Enable touch probe 1 |
| 1 | 0 | Trigger first event |
| | 1 | Continuous |
| 3, 2 | 00b* | Trigger with touch probe 1 input |
| | 01b | Trigger with zero impulse signal or position encoder |
| | 10b | Touch probe source as defined in object 60D0h, sub-index 01h |
| | 11b | reserved |
| 4 | 0 | Switch off sampling at positive edge of touch probe 1 |
| | 1 | Enable sampling at positive edge of touch probe 1 |
| 5 | 0 | Switch off sampling at negative edge of touch probe 1 |
| | 1 | Enable sampling at negative edge of touch probe 1 |
| 6, 7 | - | User-defined (e.g. for testing) |
| 8 | 0 | Switch off touch probe 2 |
| | 1 | Enable touch probe 2 |
| 9 | 0 | Trigger first event |
| | 1 | continuous |
| 11, 10 | 00b | Trigger with touch probe 2 input |
| | 01b | Trigger with zero impulse signal or position encoder |
| | 10b | Touch probe source as defined in object 60D0h, sub-index 02h |
| | 11b | reserved |
| 12 | 0 | Switch off sampling at positive edge of touch probe 2 |
| | 1 | Enable sampling at positive edge of touch probe 2 |
| 13 | 0 | Switch off sampling at negative edge of touch probe 2 |
| | 1 | Enable sampling at negative edge of touch probe 2 |
| 14, 15 | - | User-defined (e.g. for testing) |

* b = binary

If both edges are selected at the same time (bit 4=1 and bit 5=1 for probe 1 or bit 12=1 and bit 13=1 for probe 2), the first edge (positive or negative) triggers the probe function. The position, latched at this edge, is taken over for both edges (positive and negative).

19.33.3.79 Object 60B9h: Touch probe status

This object indicates the status of the touch probe.

| | |
|---------------|--------------------|
| Index | 60B9h |
| Name | Touch probe status |
| Object code | Variable |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | UNSIGNED16 |
| Default value | 0 |

Definition of the status:

| Bit | Value | Meaning |
|----------|-------|---|
| 0 | 0 | Touch probe 1 is switched off |
| | 1 | Touch probe 1 is enabled |
| 1 | 0 | Touch probe 1 no positive edge value stored |
| | 1 | Touch probe 1 positive edge position stored |
| 2 | 0 | Touch probe 1 no negative edge value stored |
| | 1 | Touch probe 1 negative edge position stored |
| 3 to 5 | 0 | reserved |
| 6, 7 | - | User-defined (e.g. for testing) |
| 8 | 0 | Touch probe 2 is switched off |
| | 1 | Touch probe 2 is enabled |
| 9 | 0 | Touch probe 2 no positive edge value stored |
| | 1 | Touch probe 2 positive edge position stored |
| 10 | 0 | Touch probe 2 no negative edge value stored |
| | 1 | Touch probe2 negative edge position stored |
| 11 to 13 | 0 | reserved |
| 14, 15 | - | User-defined (e.g. for testing) |

19.33.3.80 Object 60BAh: Touch probe 1 positive edge

This object provides the position value of the touch probe 1 at positive edge.

| | |
|---------------|-----------------------------|
| Index | 60BAh |
| Name | Touch probe 1 positive edge |
| Object code | Variable |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | INTEGER32 |
| Default value | no |

19.33.3.81 Object 60BBh: Touch probe 1 negative edge

This object provides the position value of the touch probe 1 at negative edge.

| | |
|---------------|-----------------------------|
| Index | 60BBh |
| Name | Touch probe 1 negative edge |
| Object code | Variable |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | INTEGER32 |
| Default value | no |

19.33.3.82 Object 60BCh: Touch probe 2 positive edge

This object provides the position value of the touch probe 2 at positive edge.

| | |
|---------------|-----------------------------|
| Index | 60BCh |
| Name | Touch probe 2 positive edge |
| Object code | Variable |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | INTEGER32 |
| Default value | no |

19.33.3.83 Object 60BDh: Touch probe 2 negative edge

This object provides the position value of the touch probe 2 at negative edge.

| | |
|---------------|-----------------------------|
| Index | 60BDh |
| Name | Touch probe 2 negative edge |
| Object code | Variable |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | INTEGER32 |
| Default value | no |

19.33.3.84 Object 60C0h: Interpolation sub mode select

In the AKD PDMM, linear interpolation between position setpoints is supported.

| | |
|---------------|-------------------------------|
| Index | 60C0h |
| Name | Interpolation sub mode select |
| Object code | VAR |
| Data type | INTEGER16 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | 0 |
| Default value | 0 |

Value description

| Value(decimal) | Description |
|----------------|--|
| 0 | Linear interpolation with a constant time. |

19.33.3.85 Object 60C1h: Interpolation data record

In the AKD PDMM, a single setpoint (target position, Subindex 1) is supported for the linear interpolation. After the last item of an interpolation data record is written to the devices input buffer, the buffer pointer is automatically incremented to the next buffer.

| | |
|---------------|--|
| Index | 60C1h |
| Name | Interpolation data record |
| Object code | ARRAY |
| Data type | INTEGER32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 1 |
| Default value | 1 |
| Subindex | 1 |
| Description | Interpolation target position in counts, the first parameter of interpolation function |
| Category | mandatory |
| Access | R/W |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Default value | no |

NOTE

A set-point value of the Interpolation data record is only taken, if beside the state machine state "Operation Enable" also the bit 4 of the DS402 controlword (Enable Interpolation, see "Object 6040h: Control word (DS402)" (→ p. 1048)) is set.

19.33.3.86 Object 60C2h: Interpolation time period

The interpolation time period is used for the PLL (phase locked loop) synchronized position modes. The unit (subindex 1) of the time is given in 10^{interpolation time index} seconds.

Only multiples of 1 ms are allowed. The two values define the internal AKD PDMM parameter FBUS.SAMPLEPERIOD (given in multiples of 62.5 Mikroseconds). Both values must be written to set a new interpolation time period. FBUS.SAMPLEPERIOD will only be updated then.

| | |
|---------------|--|
| Index | 60C2h |
| Name | Interpolation time period |
| Object code | RECORD |
| Data type | Interpolation time period record (0080h) |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported, FBUS.SAMPLEPERIOD |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | 2 |
| Subindex | 1 |
| Description | Interpolation time units |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | 2 |
| Subindex | 2 |
| Description | Interpolation time index |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER8 |
| Default value | -3 |

19.33.3.87 Object 60C4h: Interpolation data configuration

In the AKD PDMM, for linear interpolation, only the value 1 in Subindex 5 is possible. For cubic interpolation, Subindex 1 and Subindex 2 is possible for manage the buffer of cubic interpolation.

| | |
|---------------|---|
| Index | 60C4h |
| Name | Interpolation data configuration |
| Object code | RECORD |
| Data type | Interpolation data configuration record (0081h) |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 6 |
| Default value | 6 |
| Subindex | 1 |
| Description | Maximum buffer size |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 10 |
| Subindex | 2 |
| Description | Actual buffer size |
| Category | mandatory |
| Access | R/O |
| PDO mapping | possible |
| Value range | 0 to 9 |
| Default value | 9 |
| Subindex | 3 |
| Description | Buffer organization |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGND8 |
| Default value | 0 |

| | |
|---------------|---------------------|
| Subindex | 4 |
| Description | Buffer position |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED16 |
| Default value | 0 |
| Subindex | 5 |
| Description | Size of data record |
| Category | mandatory |
| Access | W |
| PDO mapping | not possible |
| Value range | 1 to 254 |
| Default value | 1 |
| Subindex | 6 |
| Description | Buffer clear |
| Category | mandatory |
| Access | W |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | 0 |

19.33.3.88 Object 60D0h: Touch probe source

This object provides the source of the touch probe function, when the dedicated bits 2/3 or 10/11 of the touch probe function (object 60B8h) are set accordingly.

| | |
|---------------|-----------------------------|
| Index | 60D0h |
| Name | Touch probe source |
| Object code | Array |
| Data type | Integer 16 |
| Category | optional |
| Subindex | 0 |
| Description | Highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | 2 |
| Subindex | 1 |
| Description | Touch probe 1 source |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | -11 to -1, 1 to 5 |
| Default value | 1 |
| Subindex | 2 |
| Description | Touch probe 2 source |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | -11 to -1, 1 to 5 |
| Default value | 1 |

Value description:

| Value | Description | Value | Description |
|-----------|---|-------|---------------------|
| 1 | Touch Probe 1 Input | 3 | Touch Probe 3 Input |
| 2 | Touch Probe 2 Input | 4 | Touch Probe4 Input |
| -1 to -11 | AKD PDMM Input related to CAPx.TRIGGER 0 to 10 | | |

19.33.3.89 Object 60E0h: Positive Torque Limit Value

The object gives the configured maximum motor torque in positive direction. The value is given per thousand (1 ‰) of rated torque.

| | |
|---------------|--|
| Index | 60E0h |
| Name | Positive Torque Limit Value |
| Object code | Variable |
| Data type | UIINTEGER16 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | UIINTEGER16 (limited by DRV.IPEAK and MOTOR.IPEAK) |
| Default value | 0 |

19.33.3.90 Object 60E1h: Negative Torque Limit Value

The object gives the configured maximum motor torque in negative direction. The value is given per thousand (1 ‰) of rated torque.

| | |
|---------------|--|
| Index | 60E1h |
| Name | Negative Torque Limit Value |
| Object code | Variable |
| Data type | UIINTEGER16 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | UIINTEGER16 (limited by DRV.IPEAK and MOTOR.IPEAK) |
| Default value | 0 |

19.33.3.91 Object 60E4h: Additional position actual value

This object provides the additional position actual values. The values are given in user-defined position units. The value is calculated analog to the calculation for the actual position 6064h via object 6091h and 6092h, but for this with the factors given by the objects 60E8h, 60E9h, 60EDh and 60EEh.

| | |
|---------------|--------------------------------------|
| Index | 60E4h |
| Name | Additional position actual value |
| Object code | ARRAY |
| Data type | INTEGER32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 |
| Description | 1st additional position actual value |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | 1 |
| Subindex | 2 |
| Description | reserved |
| Category | |
| Access | |
| PDO mapping | |
| Value range | |
| Default value | |
| Subindex | 3 |
| Description | 3rd additional position actual value |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | 0 |

19.33.3.92 Object 60E8h: Additional gear ratio – motor shaft revolutions

This object provides the motor shaft revolutions for the additional gear ratio calculation. This object shall be used with the corresponding subindex of the object 60EDh (driving shaft revolutions for the additional gear ratio calculation). The value of the object 60E4h is calculated analog to the gear ration calculation for the actual position 6064h via object 6091h.

| | |
|---------------|---|
| Index | 60E8h |
| Name | Additional gear ratio – motor shaft revolutions |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 |
| Description | 1st additional gear ratio - motor shaft |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |
| Subindex | 2 |
| Description | reserved |
| Category | |
| Access | |
| PDO mapping | |
| Value range | |
| Default value | |
| Subindex | 3 |
| Description | 3rd additional gear ratio - motor shaft |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |

19.33.3.93 Object 60E9h: Additional feed constant – feed

This object provides the feed for the additional feed constant calculation. This object shall be used with the corresponding subindex of the object 60EEh (driving shaft revolutions for the additional feed constant calculation). The value of the object 60E4h is calculated analog to the feed constant calculation for the actual position 6064h via object 6092h.

| | |
|---------------|-------------------------------------|
| Index | 60E9h |
| Name | Additional feed constant – feed |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 |
| Description | 1st additional feed constant – feed |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |
| Subindex | 2 |
| Description | reserved |
| Category | |
| Access | |
| PDO mapping | |
| Value range | |
| Default value | |
| Subindex | 3 |
| Description | 3rd additional feed constant – feed |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |

19.33.3.94 Object 60EDh: Additional gear ratio – driving shaft revolutions

This object provides the driving shaft revolutions for the additional gear ratio calculation. This object shall be used with the corresponding subindex of the object 60E8h (motor shaft revolutions for the additional gear ratio calculation). The value of the object 60E4h is calculated analog to the gear ration calculation for the actual position 6064h via object 6091h.

| | |
|---------------|---|
| Index | 60EDh |
| Name | Additional gear ratio – driving shaft revolutions |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 |
| Description | 1st additional gear ratio - driving shaft |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |
| Subindex | 2 |
| Description | reserved |
| Category | |
| Access | |
| PDO mapping | |
| Value range | |
| Default value | |
| Subindex | 3 |
| Description | 3rd additional gear ratio - driving shaft |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |

19.33.3.95 Object 60EEh: Additional feed constant - driving shaft revolutions

This object provides the driving shaft revolutions for the additional feed constant calculation. This object shall be used with the corresponding subindex of the object 60E9h (feed for the additional feed constant calculation). The value of the object 60E4h is calculated analog to the feed constant calculation for the actual position 6064h via object 6092h.

| | |
|---------------|--|
| Index | 60EEh |
| Name | Additional feed constant - driving shaft revolutions |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 |
| Description | 1st additional feed constant – driving shaft revolutions |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |
| Subindex | 2 |
| Description | reserved |
| Category | |
| Access | |
| PDO mapping | |
| Value range | |
| Default value | |
| Subindex | 3 |
| Description | 3rd additional feed constant – driving shaft revolutions |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |

19.33.3.96 Object 60F4h: Following error actual value (DS402)

This object returns the current value of the following error in units defined by the user.

| | |
|---------------|------------------------------|
| Index | 60F4h |
| Name | Following error actual value |
| Object code | VAR |
| Data type | Integer32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Default value | 0 |

19.33.3.97 Object 60FCh: Position demand internal value (DS402)

This object provides the output of the trajectory generator in position modes. The value is consistent in scaling to the actual internal position value (6063h) and the first setpoint in object 60C1h.

| | |
|---------------|--------------------------------|
| Index | 60FCh |
| Name | Position demand internal value |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Default value | 0 |

19.33.3.98 Object 60FDh: Digital inputs (DS402)

This index defines simple digital inputs for drives. The manufacturer bits 16 to 2216 to 18 display the actual state of the digital inputs 1 to 71 to 3 (DINx.STATE). The manufacturer bits 24 to 3024 to 26 latch a state change of the digital inputs 1 to 71 to 3. Bits 24 to 3024 to 26 can be reset with object"20B8h" (→ p. 1040).

| | |
|---------------|----------------|
| Index | 60FDh |
| Name | digital inputs |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | UNSIGNED32 |
| Default value | 0 |

| | | | | | | | |
|-----------------------|----|----------|---|--------|-------------|-------------------|-------------------|
| 31 | 16 | 15 | 4 | 3 | 2 | 1 | 0 |
| manufacturer specific | | reserved | | enable | home switch | pos. limit switch | neg. limit switch |
| MSB | | | | | | | LSB |

19.33.3.99 Object 60FEh: Digital outputs (DS402)

This index defines simple digital outputs for drives. The manufacturer bits 16 and 17 are bit 16 is show the actual status of the digital outputs 1 and 2 output 1.

| | | | | | | |
|-----------------------|-----------------------------|-------|-------|----------|---|-----------|
| Index | 60FEh | | | | | |
| Name | digital outputs | | | | | |
| Object code | Array | | | | | |
| Data type | UNSIGNED32 | | | | | |
| Category | optional | | | | | |
| Subindex | 0 | | | | | |
| Description | highest sub-index supported | | | | | |
| Category | mandatory | | | | | |
| Access | R/O | | | | | |
| PDO mapping | not possible | | | | | |
| Value range | 2 | | | | | |
| Default value | 2 | | | | | |
| Subindex | 1 | | | | | |
| Description | physical outputs | | | | | |
| Category | mandatory | | | | | |
| Access | R/W | | | | | |
| PDO mapping | possible | | | | | |
| Value range | UNSIGNED32 | | | | | |
| Default value | 0 | | | | | |
| Subindex | 2 | | | | | |
| Description | bit mask | | | | | |
| Category | optional | | | | | |
| Access | R/W | | | | | |
| PDO mapping | not possible | | | | | |
| Value range | UNSIGNED32 | | | | | |
| Default value | 0 | | | | | |
| 31 | 18 | 17 | 16 | 15 | 1 | 0 |
| manufacturer specific | | DOUT2 | DOUT1 | reserved | | set brake |
| MSB | | | | | | LSB |

19.33.3.100 Object 60FFh: Target velocity (DS402)

The speed setpoint (target velocity) represents the setpoint for the ramp generator.

| | |
|---------------|-----------------------------|
| Index | 60FFh |
| Name | target velocity, VL.CMDU |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | pv |
| Access | R/W |
| PDO mapping | possible |
| Unit | increments |
| Value range | (-2^{31}) to $(2^{31}-1)$ |
| Default value | — |
| Float scaling | 1000:1 |
| EEPROM | no |

19.33.3.101 Object 6502h: Supported drive modes (DS402)

A drive can support more than one and several distinct modes of operation. This object gives an overview of the implemented operating modes in the device. This object is read only.

| | |
|---------------|------------------------|
| Index | 6502h |
| Name | supported drive modes |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | optional |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0xE5 (csp ip hm pv pp) |

| | | | | | | | | | | | | | | |
|--------------------------|----------|-------|-----|-----|-----|----|----|----------|----|----|----|----|---|-----|
| 31 | 16 | 15 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| manufacturer specific | reserved | cstca | cst | csv | csp | ip | hm | reserved | tq | pv | vl | pp | | |
| MSB | | | | | | | | | | | | | | LSB |

20 EtherCAT Profile

| | |
|--|------------|
| 20.1 Slave Register | 933 |
| 20.2 AL Event (Interrupt Event) and Interrupt Enable | 934 |
| 20.3 Phase Run-Up | 936 |
| 20.4 CANopen over EtherCAT (CoE) State Machine | 939 |
| 20.5 Fixed PDO Mappings | 943 |
| 20.6 Flexible PDO Mappings | 945 |
| 20.7 Supported Cyclical Setpoint and Actual Values | 951 |
| 20.8 Supported Operation Modes | 953 |
| 20.9 Adjusting EtherCAT Cycle Time | 953 |
| 20.10 Maximum Cycle Times depending on operation mode | 953 |
| 20.11 Synchronization | 955 |
| 20.12 Latch Control Word and Latch Status Word | 956 |
| 20.13 Mailbox Handling | 958 |
| 20.14 Important Configuration Parameters | 964 |
| 20.15 EEPROM Content | 965 |

20.1 Slave Register

The table below gives the addresses of individual registers in the FPGA memory. The data is provided in little-endian format, with the 'least significant byte' occupying the lowest address. A detailed description of all registers and FPGA memory locations is available in the "EtherCAT Slave Controller" description of the EtherCAT user organization (www.EtherCAT.org).

| Address | Length (Byte) | Description | ZA ECAT* | ZA Drive* |
|---------|---------------|---|----------|-----------|
| 0x0120 | 2 | AL Control | R/W | R/O |
| 0x0130 | 2 | AL Status | R/O | R/W |
| 0x0134 | 2 | AL Status Code | R/O | R/W |
| 0x0204 | 2 | Interrupt Enable Register | R/O | R/W |
| 0x0220 | 2 | AL Event (IRQ Event) | R/W | R/O |
| 0x0800 | 8 | Sync Manager 0 (Mail Out Control Register) | R/W | R/O |
| 0x0808 | 8 | Sync Manager 1 (Mail In Control Register) | R/W | R/O |
| 0x0810 | 8 | Sync Manager 2 (Process data Output Control Register) | R/W | R/O |
| 0x0818 | 8 | Sync Manager 3 (Process data Input Control Register) | R/W | R/O |
| 0x0820 | 8 | Sync Manager 4 | R/W | R/O |
| 0x0828 | 8 | Sync Manager 5 | R/W | R/O |
| 0x0830 | 8 | Sync Manager 6 | R/W | R/O |
| 0x0838 | 8 | Sync Manager 7 | R/W | R/O |
| 0x0840 | 8 | Sync Manager 8 | R/W | R/O |
| 0x1100 | Max. 64 | ProOut Buffer (Process data Output, set-points ECAT) | R/W | R/O |
| 0x1140 | Max. 64 | ProIn (Process data Input, act. values ECAT) | R/O | R/W |
| 0x1800 | 512 | Mail Out Buffer (Object Channel Buffer ECAT, byte-length is specified in the device description file) | R/W | R/O |
| 0x1C00 | 512 | Mail In Buffer (Object Channel Buffer Drive, byte-length is specified in the device description file) | R/O | R/W |

* ZA ECAT = Access mode EtherCAT

* ZA Drive = Access mode drive

20.2 AL Event (Interrupt Event) and Interrupt Enable

Communication between the drive and the EtherCAT FPGA can be interrupt-driven. The interrupt enable register and the AL event register are responsible for the EtherCAT interface interrupt functionality.

There are two events which lead also to a HW interrupt within the drive, the EEPROM emulation event and the SyncManager 2 event. The actual values of the drive (SyncManager 3 data) are written without any AL event request during each HW IRQ, e.g. triggered by a SyncManager 2 event. The Mailbox exchange between the master and the AKD PDMM is completely handled by polling the AL event register within the background task of the drive.

The drive activates individual EtherCAT interface events when the corresponding bit of the interrupt enable register is set to 1. When it is set to 0, the hardware interrupts for the specific events are deactivated.

20.2.1 Interrupt Enable Register (Address 0x0204:0x0205)

| Parameter | Address | Bit | ZA Drive | ZA ECAT | Description |
|--|---------|--------|----------|---------|---|
| AL Control Event | 0x204 | 0 | R/W | R/O | Activation of AL control event for phase run-up |
| - | 0x204 | 1 | R/W | R/O | Reserved |
| Sync0 DC Distributed Clock | 0x204 | 2 | R/W | R/O | Activation of distributed clock (DC) sync 0 interrupts for entire communication |
| Sync1 DC Distributed Clock | 0x204 | 3 | R/W | R/O | Activation of distributed clock (DC) sync 1 interrupts for entire communication |
| SyncManager activation register change | 0x204 | 4 | R/W | R/O | Activation of 'SyncManager activation register change' IRQ. |
| EEPROM emulation event | 0x204 | 5 | R/W | R/O | Activation of the EEPROM emulation interrupts. |
| - | 0x204 | 3 to 7 | R/W | R/O | Reserved |
| Sync Manager 0 Event (Mail Out Event) | 0x205 | 0 | R/W | R/O | Activation of output event mailbox (SDO, Sync Manager 0) for object channel. |
| Sync Manager 1 Event (Mail In Event) | 0x205 | 1 | R/W | R/O | Activation of input event mailbox (SDO, Sync Manager 1) for object channel. |
| Sync Manager 2 Event (Pro Out Event) | 0x205 | 2 | R/W | R/O | Activation of output event process data (PDO, card's cyclical setpoints) |
| Sync Manager 3 Event (Pro In Event) | 0x205 | 3 | R/W | R/O | Activation of input event process data (PDO, drive's cyclical actual values) |
| - | 0x205 | 4 to 7 | R/W | R/O | Reserved |

20.2.2 AL Event Request (Address 0x0220:0x0221)

When the relevant bit of the AL event request register is set to 1, the EtherCAT interface tells the drive which event it should process by the AKD PDMM.

| Parameter | Address | Bit | ZA Drive | ZA ECAT | Description |
|--|---------|--------|----------|---------|---|
| AL Control Event | 0x220 | 0 | R/O | R/W | Processing of AL control event for phase run-up |
| Sync0 Distributed Clock (DC) Event | 0x220 | 2 | R/O | R/W | Processing of a distributed clock (DC) event |
| Sync1 Distributed Clock (DC) Event | 0x220 | 3 | R/O | R/W | Processing of a distributed clock (DC) event |
| SyncManager activation register change | 0x220 | 4 | R/O | R/W | The content of the SyncManager activation register has been changed. |
| EEPROM emulation event | 0x220 | 5 | R/O | R/W | Processing of an EEPROM emulation event in order to identify the AKD PDMM within the network. |
| - | 0x220 | 6 to 7 | R/O | R/W | Reserved |
| Sync Manager 0 Event | 0x221 | 0 | R/O | R/W | Mailbox request (SDO, Sync Manager 0) for object channel. |
| Sync Manager 1 Event | 0x221 | 1 | R/O | R/W | Mailbox response (SDO, Sync Manager 1) for object channel. |
| Sync Manager 2 Event | 0x201 | 2 | R/O | R/W | Process data output (PDO, card's cyclical setpoints) |
| Sync Manager 3 Event | 0x201 | 3 | R/O | R/W | Process data input (PDO, drive's cyclical actual values) |
| Sync Manager 4 – | | | | | |
| Sync Manager 7 Event | 0x221 | 4 to 7 | R/O | R/W | Reserved |
| Sync Manager 8 – | | | | | |
| Sync Manager 15 Event | 0x222 | 0 to 7 | R/O | R/W | Reserved |

20.3 Phase Run-Up

The AL control, AL status and AL status code registers are responsible for communication phase run-up (also referred to as EtherCAT status change), for current status display and for any fault messages. The drive responds to every EtherCAT interface transition request made by the AL control register via the AL Status and AL Status Code registers. Any fault messages are displayed in the AL status code register.

A status change within the AL control register is polled within the AKD PDMM, which means that an AL control event does not lead to a HW interrupt within the drive.

20.3.1 AL Control (Address 0x0120:0x0121)

| Parameter | Address | Bit | ZA Drive | ZA ECAT | Description |
|--------------------------------|---------|---------|----------|---------|---|
| Status | 0x120 | 3 to 0 | R/O | W/O | 0x01: Init Request |
| 0x02: PreOperational Request | | | | | |
| 0x03: Bootstrap Mode Request | | | | | |
| 0x04: Safe Operational Request | | | | | |
| 0x08: Operational Request | | | | | |
| Acknowledgement | 0x120 | 4 | R/O | W/O | 0x00: No fault acknowledgement 0x01: Fault acknowledgement (positive edge) |
| Reserved | 0x120 | 7 to 5 | R/O | W/O | - |
| Applic. specific | 0x120 | 15 to 8 | R/O | W/O | - |

20.3.2 AL Status (Address 0x0130:0x0131)

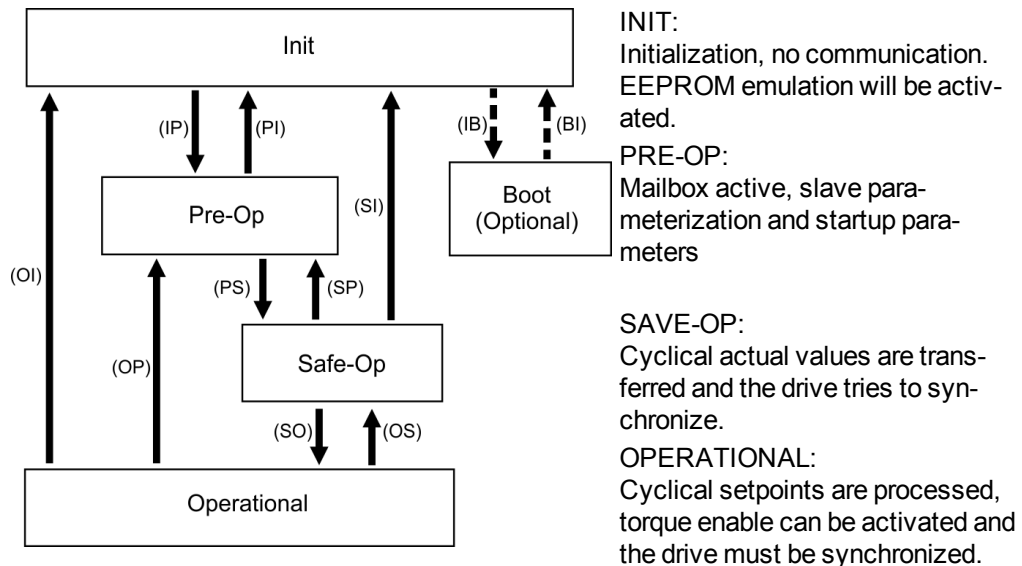
| Parameter | Address | Bit | ZA Drive | ZA ECAT | Description |
|------------------------|---------|---------|----------|---------|---|
| Status | 0x130 | 3 to 0 | W/O | R/O | 0x01: Init |
| 0x02: PreOperational | | | | | |
| 0x03: Bootstrap Mode | | | | | |
| 0x04: Safe Operational | | | | | |
| 0x08: Operational | | | | | |
| Status change | 0x130 | 4 | W/O | R/O | 0x00: Acknowledgement 0x01: Error, e.g. forbidden transition |
| Reserved | 0x130 | 7 to 5 | W/O | R/O | - |
| Applic. specific | 0x130 | 15 to 8 | W/O | R/O | - |

20.3.3 AL Status Code (Address 0x0134:0x0135)

| Parameter | Address | Bit | ZA Drive | ZA ECAT | Description |
|-----------|------------------------------------|--------|--|---------|--------------------|
| Status | 0x134 | 7 to 0 | W/O | R/O | See table below |
| Status | 0x135 | 7 to 0 | W/O | R/O | See table below |
| Code | Description | | Current Status (Status change) | | Resulting Status |
| 0x0000 | No error | | All | | Current Status |
| 0x0011 | Invalid requested state change | | I -> S, I -> O, P -> O, O -> B, S -> B, P -> B | | Current Status + E |
| 0x0017 | Invalid sync manager configuration | | I -> P, P -> S | | Current Status + E |

No other codes are supported.

20.3.4 EtherCAT communication phases

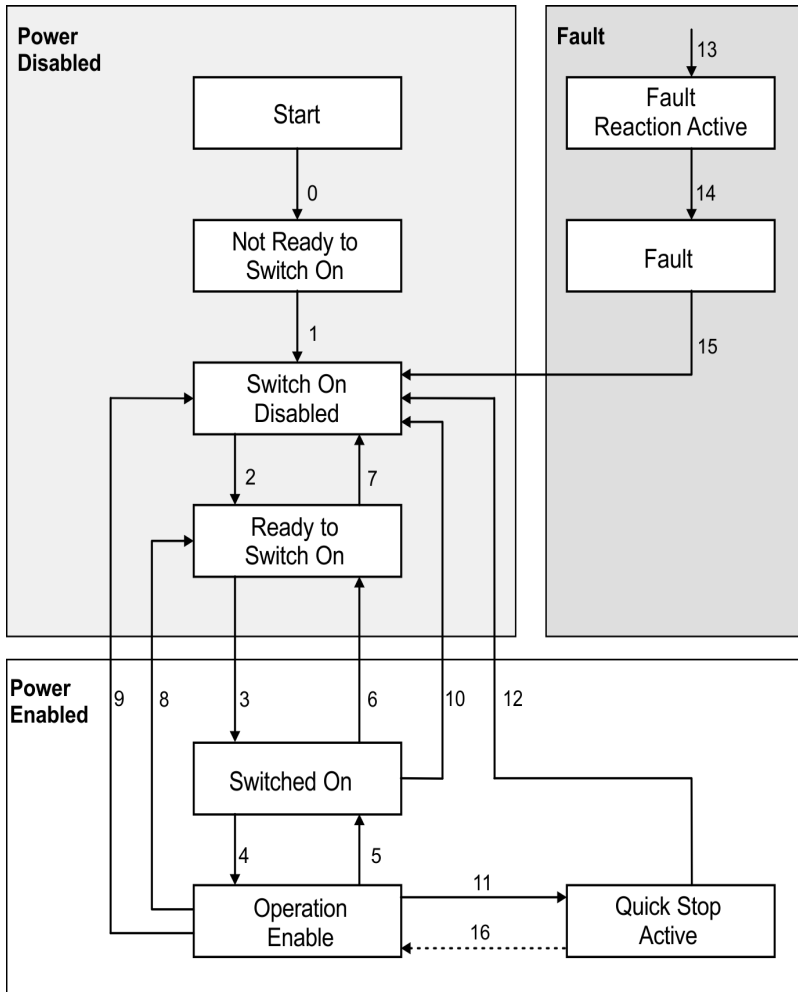


Individual communication transitions

| Transition | AL Control (Bit 3 to 0) | Description |
|------------|-------------------------|--|
| (IB) | 0x03 | - |
| (BI) | - | - |
| (IP) | 0x02 | AKD PDMM reads the SyncManager 0 & 1 configuration and verifies the value of the start-address and the length. The AKD PDMM prepares itself for handling SyncManager 0 events. |
| (PI) | 0x01 | - |
| (PS) | 0x04 | AKD PDMM reads the SyncManager 2 & 3 configuration and verifies the value of the start-address and the length. |
| (SP) | 0x02 | - |
| (SI) | 0x01 | - |

| Transition | AL Control (Bit 3 to 0) | Description |
|------------|-------------------------|---|
| (SO) | 0x08 | The SsyncManager 2 hardware interrupt will be enabled by the drive. |
| (OS) | 0x04 | Deactivation of SsyncManager 2 hardware interrupt. |
| (OP) | 0x02 | Deactivation of SsyncManager 2 hardware interrupt.. |
| (OI) | 0x01 | Deactivation of SsyncManager 2 hardware interrupt. |

20.4 CANopen over EtherCAT (CoE) State Machine



The state machine for the control and status words corresponds to the CANopen state machine in accordance with DS402.

CANopen control and status words are captured in every instance of fixed PDO mapping (see chapter "Fixed PDO Mappings" (→ p. 943)).

20.4.1 Status Description

| Status | Description |
|------------------------|--|
| Not Ready to Switch On | The drive is not ready to switch on; the controller has not indicated readiness for service. The drive is still in the boot phase or in fault status. |
| Switch On Disable | In 'Switch On Disable' status, the amplifier cannot be enabled via the EtherCAT interface, because (for example) there is no connection to a power source. |
| Ready to Switch On | In 'Ready to Switch On' status, the drive can be enabled via the control word. |
| Switched On | In 'Switched On' status, the amplifier is enabled, but the setpoints of the EtherCAT-interface are not yet transferred. The amplifier is idle, and a positive edge in bit 3 of the control word activates setpoint transfer (transition to 'Operation Enable' status). |
| Operation Enable | In this status, the drive is enabled and setpoints are transferred from the EtherCAT interface. |
| Quick Stop Active | The drive follows a quick stop ramp. |
| Fault Reaction Active | The drive responds to a fault with an emergency stop ramp. |
| Fault | A fault is pending, the drive is stopped and disabled. |

20.4.2 Commands in the Control Word

Bit assignment in the control word

| Bit | Name | Bit | Name |
|-----|---|-----|-----------------------|
| 0 | Switch on | 8 | Pause/halt |
| 1 | Disable Voltage | 9 | reserved |
| 2 | Quick Stop | 10 | reserved |
| 3 | Enable Operation | 11 | reserved |
| 4 | Operation mode specific | 12 | reserved |
| 5 | Operation mode specific | 13 | Manufacturer-specific |
| 6 | Operation mode specific | 14 | Manufacturer-specific |
| 7 | Reset Fault (only effective for faults) | 15 | Manufacturer-specific |

Commands in the control word

| Command | Bit 7 Fault Reset | Bit 3 Enable Operation | Bit 2 Quick Stop | Bit 1 Disable Voltage | Bit 0 Switch On | Transitions |
|-------------------|-------------------------|------------------------------|------------------------|-----------------------------|-----------------------|--------------|
| Shutdown | X | X | 1 | 1 | 0 | 2, 6, 8 |
| Switch on | X | X | 1 | 1 | 1 | 3 |
| Disable Voltage | X | X | X | 0 | X | 7, 9, 10, 12 |
| Quick Stop | X | X | 0 | 1 | X | 7, 10, 11 |
| Disable Operation | X | 0 | 1 | 1 | 1 | 5 |
| Enable Operation | X | 1 | 1 | 1 | 1 | 4, 16 |
| Fault Reset | 1 | X | X | X | X | 15 |

Bits labeled **X** are irrelevant. **0** and **1** indicate the status of individual bits.

Mode-dependent bits in the control word

The following table shows the mode-dependent bits in the control word. Only manufacturer-specific modes are supported at present. The individual modes are set by Object 6060h Modes of operation.

| Operation mode | No | Bit 4 | Bit 5 | Bit 6 |
|----------------------------------|-----|------------------------|------------------------|-------------------|
| Profile Position Mode (pp) | 01h | new_setpoint | change_set_immediately | absolute/relative |
| Profile Velocity Mode (pv) | 03h | reserved | reserved | reserved |
| Profile Torque Mode (tq) | 04h | reserved | reserved | reserved |
| Homing Mode (hm) | 06h | homing_operation_start | reserved | reserved |
| Interpolated Position Mode (ip) | 07h | | reserved | reserved |
| Cyclic synchronous position mode | 08h | reserved | reserved | reserved |

Description of the remaining bits in the control word

Bit 8: (Pause) If Bit 8 is set, then the drive halts (pauses) in all modes. The setpoints (speed for homing or jogging, motion task number, setpoints for digital mode) for the individual modes are retained.

Bit 9,10: These bits are reserved for the drive profile (DS402).

Bit 13, 14, 15: These bits are manufacturer-specific, and reserved at present.

20.4.3 State Machine Bits (status word)

Bit assignment in the status word

| Bit | Name | Bit | Name |
|-----|--------------------|-----|------------------------------------|
| 0 | Ready to switch on | 8 | Manufacturer-specific (reserved) |
| 1 | Switched on | 9 | Remote (always 1) |
| 2 | Operation enable | 10 | Target reached |
| 3 | Fault | 11 | Internal limit active |
| 4 | Voltage enabled | 12 | Operation mode specific (reserved) |
| 5 | Quick stop | 13 | Operation mode specific (reserved) |
| 6 | Switch on disabled | 14 | Manufacturer-specific (reserved) |
| 7 | Warning | 15 | Manufacturer-specific (reserved) |

States of the state machine

| State | Bit 6 switch on disable | Bit 5 quick stop | Bit 3 fault | Bit 2 operation enable | Bit 1 switched on | Bit 0 ready to switch on |
|------------------------|----------------------------------|------------------------|----------------|------------------------------|-------------------------|-----------------------------------|
| Not ready to switch on | 0 | X | 0 | 0 | 0 | 0 |
| Switch on disabled | 1 | X | 0 | 0 | 0 | 0 |
| Ready to switch on | 0 | 1 | 0 | 0 | 0 | 1 |
| Switched on | 0 | 1 | 0 | 0 | 1 | 1 |
| Operation enabled | 0 | 1 | 0 | 1 | 1 | 1 |
| Fault | 0 | X | 1 | 0 | 0 | 0 |
| Fault reaction active | 0 | X | 1 | 1 | 1 | 1 |
| Quick stop active | 0 | 0 | 0 | 1 | 1 | 1 |

Bits labeled **X** are irrelevant. **0** and **1** indicate the status of individual bits.

Description of the remaining bits in the status word

Bit 4: `voltage_enabled` The DC-link voltage is present if this bit is set.

Bit 7: `warning` There are several possible reasons for Bit 7 being set and this warning being produced. The reason for this warning can be revealed by using the Object 20subindex manufacturer warnings.

Bit 9: `remote` is always set to 1, i.e. the drive can always communicate and be influenced via the RS232 - interface.

Bit 10: `target_reached` This is set when the drive has reached the target position.

Bit 11: `internal_limit_active` This bit specifies that a movement was or is limited. In different modes, different warnings cause the bit to be set.

20.5 Fixed PDO Mappings

Various ready-to-use mappings can be selected for cyclic data exchange via SDO's of the object 0x1C12 and 0x1C13. Using object 0x1C12 subindex 1 (Sync Manager 2 assignment), a fixed mapping for the cyclic command values can be set with the values 0x1701, 0x1702, 0x1720 to 0x1724. Using object 0x1C13 subindex 1 (Sync Manager 3 assignment), a fixed mapping for the cyclic actual values can be set via the data 0x1B01, 0x1B20 to 0x1B25.

Use the sequence below to select the fixed command value mapping 0x1701 via SDO's:

1. SDO write access to object 0x1C12Sub0 Data:0x00
2. SDO write access to object 0x1C12Sub1 Data:0x1701
3. SDO write access to object 0x1C12Sub0 Data:0x01

NOTE

Up to firmware version 1.8.x.x AKD.XML file, fixed mapping 0x1701 called out 0x6062sub0 as the "Position Command". From AKD PDMM firmware release 1.8.5.0, the AKD.XML will be changed to call out 0x60C1sub1 as the "Position Command" and an additional XML file called "AKD_TwinCAT.XML" will be added to support TwinCat 2x and older. In reality, SDO 0x6062sub0 is not supported in the AKD PDMM firmware but was called in the fixed mapping to support a TwinCat issue.

Position interface, supported fixed mappings:

| | |
|--------|--|
| 0x1701 | Position command value (4 bytes), Control word (2 bytes), total (6 bytes) |
| 0x1720 | Control Word (2 bytes), Interpolated position command value (4 bytes), Latch control word (2 bytes), Torque feed forward (2 bytes), Digital outputs (2 bytes) |
| 0x1721 | Interpolated position command value (4 bytes), Control Word (2 bytes), Torque feed forward (2 bytes) |
| 0x1722 | Control word (2 byte), Interpolated position command value (4 bytes), Latch control word (2 bytes), Torque feed forward (2 bytes), Digital outputs (2 bytes), max. torque (2 bytes) |
| 0x1723 | Control word (2 bytes), Interpolated position command value (4 bytes), Latch control word (2 bytes), Torque feed forward (2 bytes), Digital outputs (2 bytes), Reset of changed input information (2 bytes) |
| 0x1724 | Target position for cyclic synchronous position mode (4 bytes), Control word (2 byte), Torque feed forward (2 bytes) |
| 0x1B01 | Position actual value (4 bytes), Status word (2 bytes), total (6 bytes) |
| 0x1B20 | Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), velocity actual value (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position positive (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes) |
| 0x1B21 | Position Actual Internal Value (4 bytes), Status word (2 bytes) |
| 0x1B22 | Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), velocity actual value (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position negative (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes) |
| 0x1B23 | Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), velocity actual value (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position positive / negative (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes) |
| 0x1B24 | Position actual value (4 bytes), status word (2 bytes) |

| | |
|--------|---|
| 0x1B25 | Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), latch position 2 positive / negative (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position 1 positive / negative (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes) |
|--------|---|

Velocity interface, supported fixed mappings:

| | |
|--------|---|
| 0x1702 | Velocity command value (4 bytes), Control word (2 bytes), total (6 bytes) |
|--------|---|

The objects, which are mapped into the fixed PDOs can be read via the subindices 1 to n of the above indices. The number of mapped entries is available by reading subindex 0 of the above indices.

Example:

A read access to object 1702 sub 0 gives a value of 2, a read on subindex 1 gives 0x60ff0020, on subindex 2 0x60400010. The meaning of these numbers can be seen in the CANopen manual or the flexible-mapping example (→ p. 948.).

20.6 Flexible PDO Mappings

In addition to the fixed PDO mapping the so-called flexible mapping of real-time objects is possible.

NOTE

Available objects for PDO mapping are listed in the object dictionaries ("Appendix" (→ p. 968))("Appendix" (→ p. 1)). All objects with the entry "yes" in column "PDO map." can be used.

Restrictions of flexible mapping:

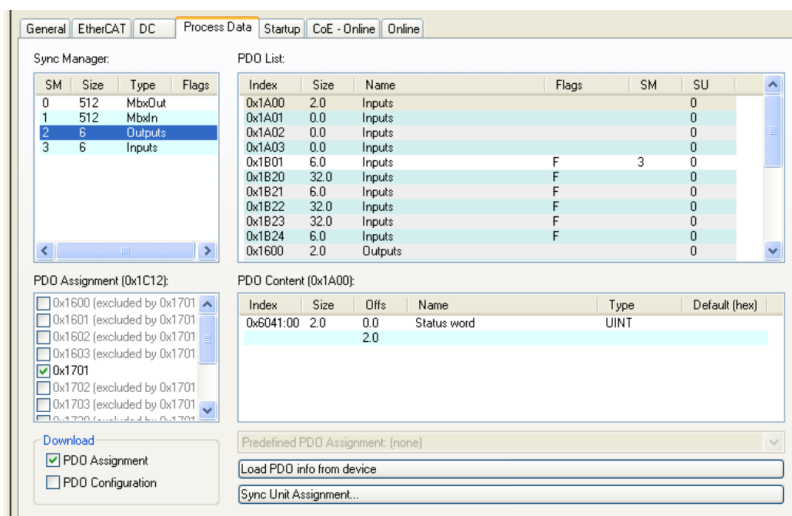
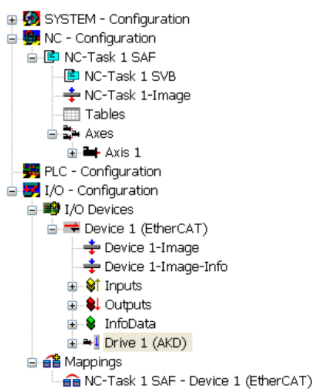
- An odd length PDO is not allowed.
- In the Rx(=set-point)-direction the dummy-object 0x0002 sub 0 with a length of 8 bits can be used to make the PDO-length even.
- In the Tx(=actual value)-direction one sub-index of the manufacturer status object 0x2002 sub 1..4 can be used to guarantee the even length of the Tx-PDO.
- These special mappings may be used if the objects 0x6060 and 0x6061 have to be used in the mapping.
- The allowed PDOs have up to 32 bytes (Tx) or 22 bytes (Rx). They are built from smaller PDO modules with a maximum length of 8 bytes. These are built by using the mapping objects 0x1600 to 0x1603 and 0x1a00 to 0x1a03.

The configuration is similar to the described sequence for the fixed mappings:

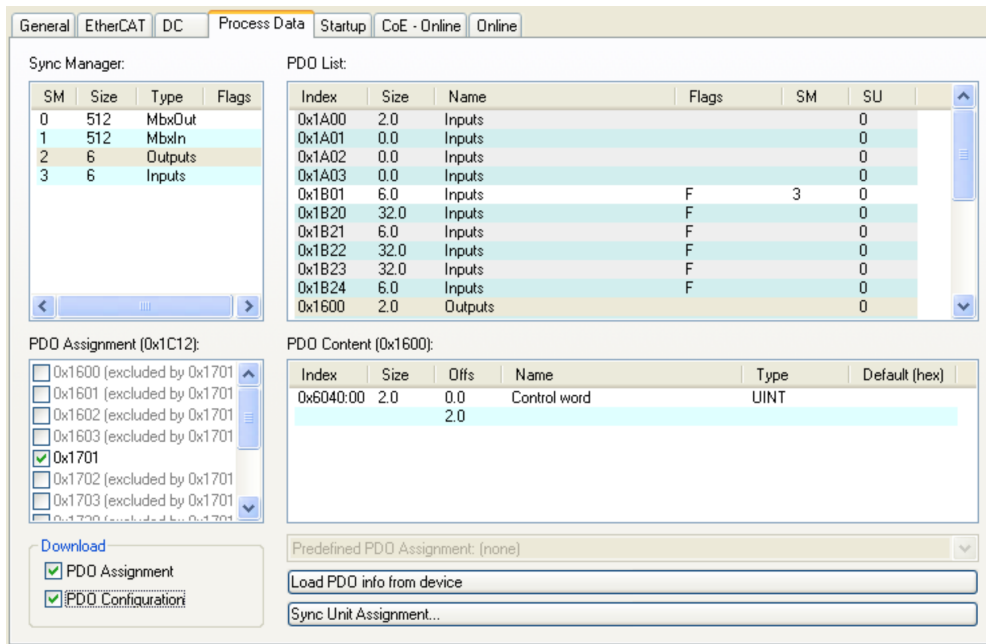
1. The mapping selection is cleared (write 0 to object 0x1C12 sub 0 and 1C13 sub 0)
2. As the AKD PDMM - implementation is based on CANopen the real-time data are build from up to 4 PDOs with 8 bytes in both directions. These PDOs are built in the same way as in a CAN-drive with the objects 0x1600 - 0x1603 and 0x1A00 - 0x1A03. Unused PDOs must be cleared with writing 0 to the subindex 0.
3. SDO write access to object 0x1C12 sub 1 .. 4 with the PDOs (0x1600 .. 0x1603), that should be used in receive direction of the AKD PDMM (set point values).
4. SDO write access to object 0x1C13 sub 1 .. 4 with the PDOs (0x1A00 .. 0x1A03), that should be used in transmit direction of the AKD PDMM (actual values).
5. SDO write access to the objects 0x1C12 sub 0 and 0x1C13 sub 0 with the number of mapped PDOs in this direction.

See an example in chapter "Flexible PDO Mappings" (→ p. 945) .

The cyclically used data are visible in the PDO-assignment window for the Inputs and Outputs of the Sync Managers. Default setting are the fixed PDOs 0x1701 and 0x1B01 (visible contents when selected in the PDO list).



If the flexible mapping is required, the PDO configuration check box must be changed.

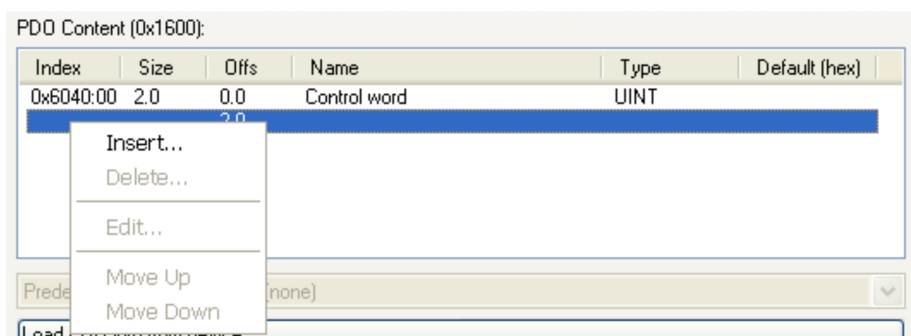


20.6.1 Example: Flexible PDO Mapping

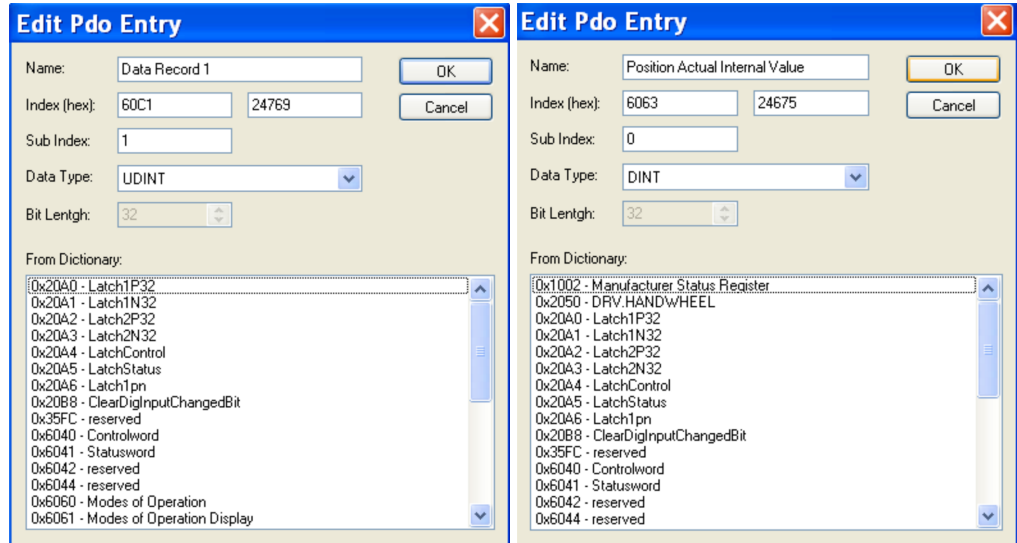
For the flexible mapping of the Outputs the fixed mapping 0x1701 has to be switched off and up to 4 free-mappable PDOs (0x1600-0x1603) can be used instead. The maximum number of bytes for each of these PDOs is 8.



After that the default mapping of e.g. the PDO 0x1600 can be extended:



A list of possible objects for the mapping will be shown and a new entry can be chosen.



In this case the setpoint for the interpolated position mode is selected.

The same is valid for the Tx-PDO-direction. Here the value of the actual internal position is selected.

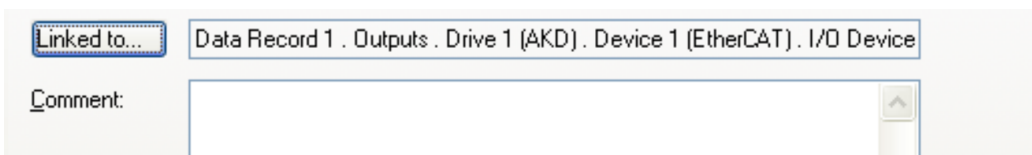
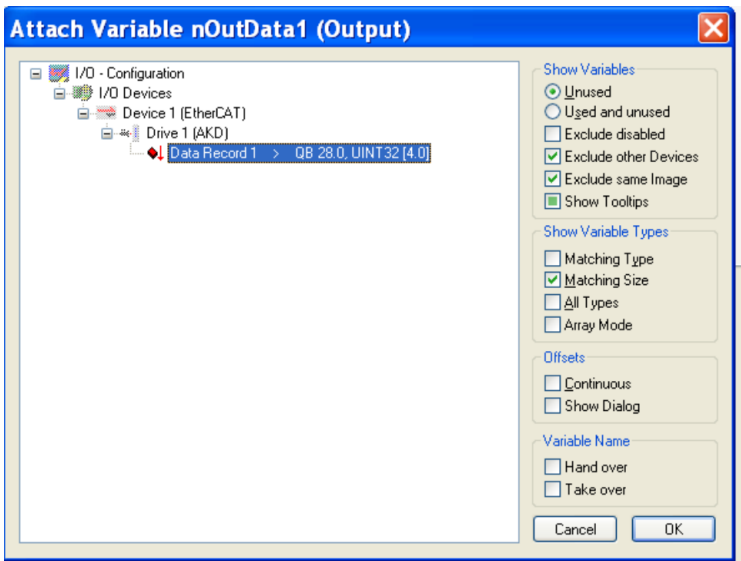
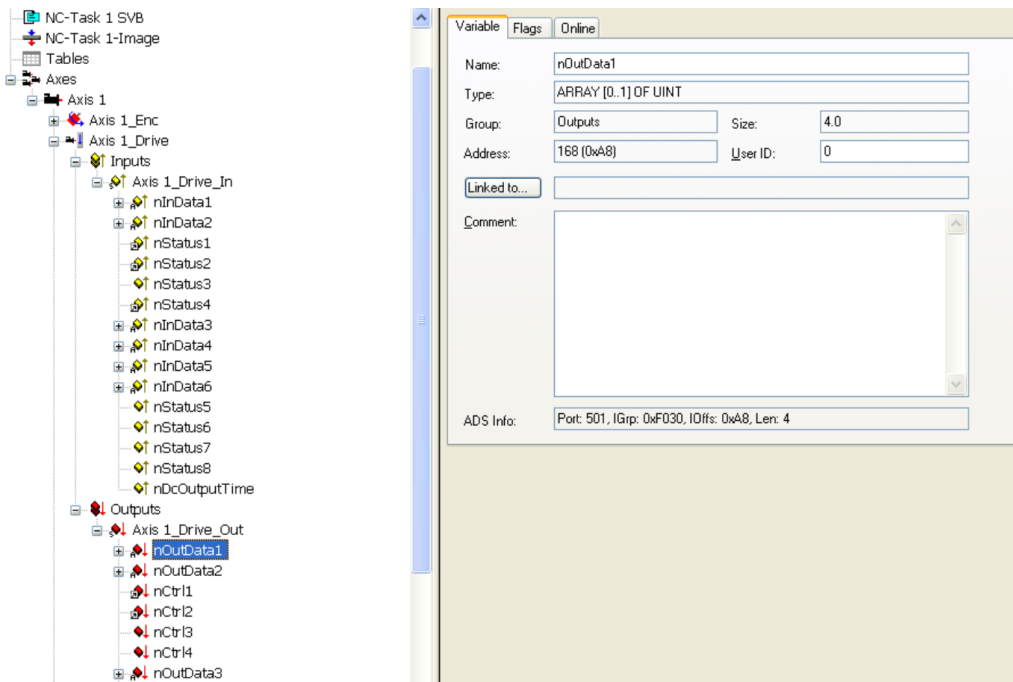
This results in the startup-SDO-list for this sample free-mapped-configuration.

| Transition | Protocol | Index | Data | Comment |
|------------|----------|-----------|-------------------------|-----------------------------|
| <PS> | CoE | 0x1C12:00 | 0x00 (0) | clear sm pdos (0x1C12) |
| <PS> | CoE | 0x1C13:00 | 0x00 (0) | clear sm pdos (0x1C13) |
| <PS> | CoE | 0x1A00:00 | 0x00 (0) | clear pdo 0x1A00 entries |
| <PS> | CoE | 0x1A00:01 | 0x60410010 (1614872592) | download pdo 0x1A00 entry |
| <PS> | CoE | 0x1A00:02 | 0x60630020 (1617100832) | download pdo 0x1A00 entry |
| <PS> | CoE | 0x1A00:00 | 0x02 (2) | download pdo 0x1A00 entr... |
| <PS> | CoE | 0x1A01:00 | 0x00 (0) | clear pdo 0x1A01 entries |
| <PS> | CoE | 0x1A02:00 | 0x00 (0) | clear pdo 0x1A02 entries |
| <PS> | CoE | 0x1A03:00 | 0x00 (0) | clear pdo 0x1A03 entries |
| <PS> | CoE | 0x1600:00 | 0x00 (0) | clear pdo 0x1600 entries |
| <PS> | CoE | 0x1600:01 | 0x60400010 (1614807056) | download pdo 0x1600 entry |
| <PS> | CoE | 0x1600:02 | 0x60C10120 (1623261472) | download pdo 0x1600 entry |
| <PS> | CoE | 0x1600:00 | 0x02 (2) | download pdo 0x1600 entr... |
| <PS> | CoE | 0x1601:00 | 0x00 (0) | clear pdo 0x1601 entries |
| <PS> | CoE | 0x1602:00 | 0x00 (0) | clear pdo 0x1602 entries |
| <PS> | CoE | 0x1603:00 | 0x00 (0) | clear pdo 0x1603 entries |
| <PS> | CoE | 0x1C12:01 | 0x1600 (5632) | download pdo 0x1C12:01 i... |
| <PS> | CoE | 0x1C12:00 | 0x01 (1) | download pdo 0x1C12 count |
| <PS> | CoE | 0x1C13:01 | 0x1B01 (6913) | download pdo 0x1C13:01 i... |
| <PS> | CoE | 0x1C13:00 | 0x01 (1) | download pdo 0x1C13 count |
| PS | CoE | 0x6060:00 | 0x07 (7) | Opmode |
| PS | CoE | 0x60C2:01 | 0x02 (2) | Cycle time |
| PS | CoE | 0x60C2:02 | 0xFD (253) | Cycle exp |

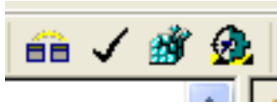
The meaning of the data (for example 0x60410010 in the mapping of 0x1A00 sub 1) is as follows:

- 0x6041 is the index of the DS402 status word
- 0x00 is the subindex of the DS402 status word
- 0x10 is the number of bits for this entry, i. e. 16 bits or 2 bytes.

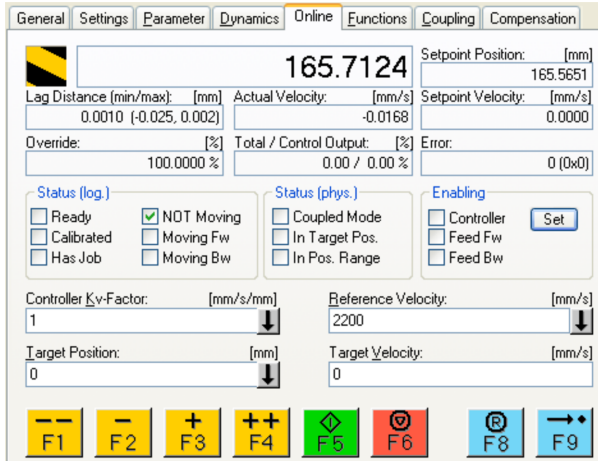
If this shall be used in the NC, the interpolation set point position has to be linked from the axis to the NC-axis.



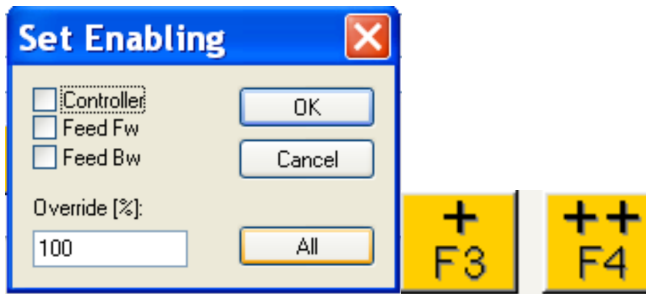
After doing this configuration the mapping can be activated as seen before in this document:



Now the NC-screen should show a position in the online window, which changes a bit in the last digits.



After enabling the power stage with the All-button, the drive can be moved via the jog-buttons or via the functions in the function menu.



20.7 Supported Cyclical Setpoint and Actual Values

Supported cyclical setpoint values

| Name | CANopen object | Data type | Description |
|----------------------------------|----------------|-----------|--|
| Target current | 0x2071 sub 0 | 32 bit | scaled in mA |
| Latch Control word | 0x20A4 sub 0 | UINT16 | |
| Clear digital Input Change Bit | 0x20B8 | 16 bit | |
| Analog output value | 0x3470 sub 3 | 16 bit | |
| External feedback position | 0x3497 sub 0 | 32 bit | |
| CANopen control-word | 0x6040 sub 0 | UINT16 | CANopen control word. |
| Modes of Operation | 0x6060 sub 0 | 8 bit | DS402 opmode setpoint |
| Velocity Window | 0x606D sub 0 | 16 bit | |
| Velocity Window Time | 0x606E sub 0 | 16 bit | |
| Target Torque | 0x6071 sub 0 | 16 bit | 0.1% resolution |
| Maximum Torque | 0x6072 sub 0 | 16 bit | |
| Target position | 0x607A sub 0 | INT32 | Used in profile position mode / cyclic synchronous position mode |
| Profile position target velocity | 0x6081 sub 0 | 32 bit | related to MT.V |
| Profile position target acc | 0x6083 sub 0 | 32 bit | related to MT.ACC |
| Profile position target dec | 0x6084 sub 0 | 32 bit | related to MT.DEC |
| Velocity feed forward | 0x60B1 sub 0 | 32 bit | |
| Torque feed forward | 0x60B2 sub 0 | INT16 | |
| Touch probe function | 0x60B8 | 16 bit | |
| Position command value | 0x60C1 sub 1 | INT32 | Interpolation data record in IP-mode |
| Digital outputs | 0x60FE sub 1 | UINT32 | |
| Velocity command value | 0x60FF sub 0 | INT32 | |

Supported cyclical actual values

| Name | CANopen object | Data type | Description |
|---------------------------------|----------------|-----------|-----------------------|
| Position actual internal value | 0x6063 sub 0 | INT32 | |
| Velocity actual value | 0x606C sub 0 | INT32 | |
| CANopen status-word | 0x6041 sub 0 | UINT16 | CANopen status word |
| Second position feedback | 0x2050 sub 0 | INT32 | |
| Digital inputs | 0x60FD sub 0 | UINT32 | |
| Following error actual value | 0x60F4 sub 0 | INT32 | |
| Latch position positive edge | 0x20A0 sub 0 | INT32 | |
| Torque actual value | 0x6077 sub 0 | INT16 | |
| Latch status | 0x20A5 sub 0 | UINT16 | |
| Actual Current | 0x2077 sub 0 | 32 bit | scaled in mA |
| Latch1 negative edge | 0x20A1 sub 0 | 32 bit | |
| Latch2 Positive | 0x20A2 sub 0 | 32 bit | |
| Latch2 Negative | 0x20A3 sub 0 | 32 bit | |
| Latch1 positive/negative edge | 0x20A6 | 32 bit | |
| Latch 2 positive/negative edge | 0x20A7 | 32 bit | |
| Modes of Operation | 0x6061 | 8 bit | DS402 opmode status |
| Position Actual Value | 0x6064 sub 0 | 32 bit | WB/ DS402 scale units |
| Touch probe status | 0x60B9 sub 0 | 16 bit | |
| Touch probe 1 positive edge pos | 0x60BA sub 0 | 32 bit | |
| Touch probe 1 negative edge pos | 0x60BB sub 0 | 32 bit | |
| Touch probe 2 positive edge pos | 0x60BC sub 0 | 32 bit | |
| Touch probe 2 negative edge pos | 0x60BD sub 0 | 32 bit | |
| Additional Pos actual value | 0x60E4 sub 0 | 48 bit | |
| Additional Pos actual value | 0x60E4 sub 1 | 32 bit | |
| Motor I2t | 0x3427 sub 3 | 32 bit | |
| Analog output value | 0x3470 sub 2 | 16 bit | |
| Analog Input value | 0x3470 sub 4 | 16 bit | |
| Manufacturer status register | 0x1002 sub 0 | 32 bit | |

20.8 Supported Operation Modes

| CANopen mode of operation | AKD PDMM mode of operation | Description |
|-----------------------------|---------------------------------|--|
| Profile velocity | DRV.OPMODE 1 DRV.CMDSOURCE 1 | 0x6060Sub0 Data: 3 In this mode, the EtherCAT master sends cyclic velocity command values to the AKD PDMM. |
| Interpolated position | DRV.OPMODE 2 DRV.CMDSOURCE 1 | 0x6060Sub0 Data: 7 In this mode of operation the EtherCAT master sends cyclic position command values to the AKD PDMM. These command values are interpolated by the AKD PDMM according to the fieldbus sample rate. |
| Homing mode | DRV.OPMODE 2 DRV.CMDSOURCE 0 | 0x6060 sub 0 data : 6 In this mode an AKD PDMM-internal homing can be done. |
| Profile Position | DRV.OPMODE 2 DRV.CMDSOURCE 0 | 0x6060sub0 Data: 1 Uses motion task 0 to execute a point to point move |
| Torque | DRV.OPMODE 0 DRV.CMDSOURCE 1 | 0x6060sub0 Data: 4 Commands torque in % of drive peak torque |
| Cyclic Synchronous Position | DRV.OPMODE 2 DRV.CMDSOURCE 1 | 0x6060sub0 Data: 8 Master calculates move profile and commands motion with position points |

20.9 Adjusting EtherCAT Cycle Time

The cycle time to be used in the drive for the cyclical setpoints and actual values can either be stored in the FBUS.SAMPLEPERIOD parameter in the amplifier or configured in the startup phase. This happens via SDO mailbox access to objects 60C2 subindex 1 and 2.

Subindex 2, known as the interpolation time index, defines the power of ten of the time value (e.g. -3 means 10⁻³ or milliseconds) while subindex 1, known as interpolation time units, gives the number of units (e.g. 4 means 4 units).

You can run a 2 ms cycle using various combinations. For example,

Index = -3, Units = 2 or

Index = -4, Units = 20 etc.

The FBUS.SAMPLEPERIOD parameter is counted in multiples of 62.5us microseconds within the device. This means, for example that 2 ms equates to FBUS.SAMPLEPERIOD value of 32.

20.10 Maximum Cycle Times depending on operation mode

The minimum cycle time for the drive is largely dependent on the drive configuration (second actual position value encoder latch functionality enabled and so on)

| Interface | Cycle time AKD PDMM |
|-----------|----------------------|
| Position | ≥ 0.25 ms (≥ 250 μs) |
| Velocity | ≥ 0.25 ms (≥ 250 μs) |
| Torque | ≥ 0.25 ms (≥ 250 μs) |

20.11 Synchronization

On all drives, the internal PLL is theoretically able to even out an average deviation of up to 4800 ppm in the cycle time provided by the master. The drive checks once per fieldbus cycle a counter within the drive internal FPGA, which is cleared by a Sync0 (Distributed clock) event. Depending of the counter value, the drive extends or decreases the 62.5 μ s MTS signal within the drive by a maximum of 300 ns.

The theoretical maximum allowed deviation can be calculated by using the following formula:

$$\max_{dev} = \frac{300[\text{ns}]}{62.5[\mu\text{s}]} \cdot 1,000,000 = 4800 \text{ [ppm]}$$

The synchronization functionality within the drive can be enabled via setting bit 0 of the FBUS.PARAM02 parameter to high. Therefore FBUS.PARAM02 must be set to the value of 1. Furthermore the distributed clock functionality must be enabled by the EtherCAT master in order to activate cyclic Sync0 events.

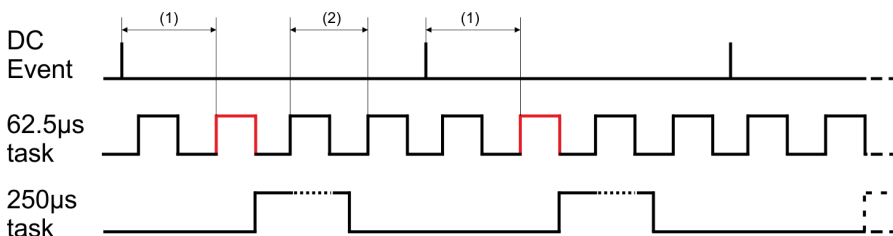
20.11.1 Synchronization behavior with distributed clocks (DC) enabled

When the EtherCAT master enables distributed clocks, a distributed clock (DC) event is created in the AKD PDMM once per fieldbus cycle. An assigned 62.5 μ s real-time task in the AKD PDMM monitors the elapsed time between the DC events and the AKD PDMM System time and extends or reduces the 62.5 μ s strobe to the CPU as necessary.

The following fieldbus parameters are used for the synchronization feature:

1. FBUS.SYNCDIST = Expected time delay of the AKD PDMM PLL-code to the DC event.
2. FBUS.SYNCACT = Actual time delay of the AKD PDMM PLL-code to the DC event.
3. FBUS.PLLTHRESH = Number of consecutive successful synchronized PLL cycles of the AKD PDMM before the Drive is considered as synchronized.
4. FBUS.SYNCWND = Synchronization window in which the AKD PDMM is considered to be synchronized. The Drive is considered synchronized as long as the following statement is true is true for FBUS.PLLTHRESH consecutive cycles:
 $\text{FBUS.SYNCDIST} - \text{FBUS.SYNCWND} < \text{FBUS.SYNCACT} < \text{FBUS.SYNCDIST} + \text{FBUS.SYNCWND}$

Example with a 4kHz fieldbus sample rate:



Explanation: The red-marked 62.5 μ s real-time task displays the AKD PDMM 62.5 μ s real-time task within one fieldbus cycle which is responsible for calling the AKD PDMM PLL-code. The time delay (1) shows the actual delay to the previous DC event, which is ideally close to the adjusted FBUS.SYNCDIST parameter. Depending on (1) the AKD PDMM slightly extends or reduce the 62.5 μ s IRQ generation of the high-priority real-time task in order to either increase or decrease the measured time delay to the DC event (1) for the next PLL cycle. The time distance (2) shows the 62.5 μ s \pm x[ms] realtime task of the AKD PDMM.

20.11.2 Synchronization behavior with distributed clocks (DC) disabled

The AKD PDMM fieldbus synchronization algorithm is similar to that used by Distributed Clocks. The difference is that the AKD PDMM synchronizes to a SyncManager2 event instead of the DC event. A SyncManager2 event is created when the EtherCAT Master sends a new package of command values to the drive while the network is in the Operational state. This occurs once per fieldbus cycle.

20.12 Latch Control Word and Latch Status Word

Latch Control word (2 Byte)

| Bit | Value (bin) | Value (hex) | Description |
|-------|----------------------|-------------|---------------------------------------|
| 0 | 00000000 00000001 | zz01 | Enable extern latch 1 (positive rise) |
| 1 | 00000000 00000010 | zz02 | Enable extern latch 1 (negative rise) |
| 2 | 00000000 00000100 | zz04 | Enable extern latch 2 (positive rise) |
| 3 | 00000000 00001000 | zz08 | Enable extern latch 2 (negative rise) |
| 4 | | | |
| 5-7 | | | Reserve |
| 8-12 | 00000001 00000000 | 01zz | Read external latch 1 (positive rise) |
| | 00000010 00000000 | 02zz | Read external latch 1 (negative rise) |
| | 00000011 00000000 | 03zz | Read external latch 2 (positive rise) |
| | 00000100 00000000 | 04zz | Read external latch 2 (negative rise) |
| 13-15 | | | Reserve |

Latch Status word (2 Byte)

| Bit | Value (bin) | Value (hex) | Description |
|-------|-------------------|-------------|--|
| 0 | 00000000 00000001 | zz01 | External latch 1 valid (positive rise) |
| 1 | 00000000 00000010 | zz02 | External latch 1 valid (negative rise) |
| 2 | 00000000 00000100 | zz04 | External latch 2 valid (positive rise) |
| 3 | 00000000 00001000 | zz08 | External latch 2 valid (negative rise) |
| 4 | | | |
| 5-7 | | | Reserve |
| 8-11 | 00000001 00000000 | z1zz | Acknowledge value external latch 1 (positive rise) |
| | 00000010 00000000 | z2zz | Acknowledge value external latch 1 (negative rise) |
| | 00000011 00000000 | z3zz | Acknowledge value external latch 2 (positive rise) |
| | 00000100 00000000 | z4zz | Acknowledge value external latch 2 (negative rise) |
| 12-15 | 00010000 00000000 | 1zzz | Zustand Digital Input 4 |
| | 00100000 00000000 | 2zzz | Zustand Digital Input 3 |

| Bit | Value (bin) | Value (hex) | Description |
|-----|-------------------|-------------|-------------------------|
| | 01000000 00000000 | 4zzz | Zustand Digital Input 2 |
| | 10000000 00000000 | 8zzz | Zustand Digital Input 1 |

20.13 Mailbox Handling

With EtherCAT, acyclical data traffic (object channel or SDO channel) is called mailbox.

NOTE

Available SDO objects are listed in the ("Appendix" (→ p. 968))("Appendix" (→ p. 1)).

This system is based around the master:

Mailbox Output:

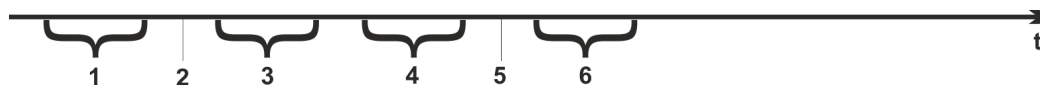
The master (EtherCAT controller) sends data to the slave (drive). This is essentially a (read/write) request from the master. Mailbox output operates via Sync Manager 0.

Mailbox Input:

The slave (drive) sends data to the master (EtherCAT controller). The master reads the slave's response. Mailbox input operates via Sync Manager 1.

Timing diagram

The timing diagram illustrates the mailbox access process:



1. The EtherCAT master writes the mailbox request to the mail-out buffer.
2. On the next interrupt, the EtherCAT interface activates a Sync Manager 0 event (mailbox output event) in the AL event register.
3. The drive reads 16 bytes from the mail-out buffer and copies them to the internal mailbox output array.
4. The drive identifies new data in the internal mailbox output array and performs an SDO access to the object requested by the EtherCAT interface. The response from the drive is written to an internal mailbox input array.
5. The drive deletes all data in the internal mailbox output array so that a new mailbox access attempt can be made.
6. The drive copies the response telegram from the internal mailbox input array to the mail-in buffer of the EtherCAT interface.

20.13.1 Mailbox Output

An interrupt by the EtherCAT-interface with a Sync Manager 0 - Event starts a Mailbox Output Process. A 1 in the Mail Out Event-Bit of the AL Event register signalizes the drive, that the EtherCAT-interface wants to send a Mailbox message and that it has already stored the required data in the Mail Out Buffer. Now 16 Byte data are read by the drive with the IRQ process. The bytes are defined as follows

| Address 0x1800 | | | | | | | | Address 0x180F | | | | | | | |
|---|---|---|---|---|---|---|------------------|--|---|----|----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| CAN over EtherCAT specific data (CoE Header) | | | | | | | | CAN specific data (standard CAN SDO) | | | | | | | |
| Byte 0 | Length of the data (Low Byte) | | | | | | | | | | | | | | |
| Byte 1 | Length of the data (High Byte) | | | | | | | | | | | | | | |
| Byte 2 | Address (Low Byte) | | | | | | | | | | | | | | |
| Byte 3 | Address (High Byte) | | | | | | | | | | | | | | |
| Byte 4 | Bit 0 to 5: Channel Bit 6 to 7: Priority | | | | | | | | | | | | | | |
| Byte 5 | Bit 0 to 3: Type | | | | | | | 1 = Reserved: ADS over EtherCAT 2 = Reserved: Ethernet over EtherCAT 3 = Can over EtherCAT...) | | | | | | | |
| | Bit 4 to 7: Reserved | | | | | | | | | | | | | | |
| Byte 6 | PDO Number (with PDO transmissions only, Bit 0 = LSB of the PDO number, see Byte 7 for MSB) | | | | | | | | | | | | | | |
| Byte 7 | Bit 0: MSB of the PDO number, see Byte 6 | | | | | | | | | | | | | | |
| | Bit 1 to 3: Reserved | | | | | | | | | | | | | | |
| | Bit 4 to 7: CoE specific type | | | | | | | 0: Reserved | | | | | | | |
| | | | | | | | | 1: Emergency message | | | | | | | |
| | | | | | | | | 2: SDO request | | | | | | | |
| | | | | | | | | 3: SDO answer | | | | | | | |
| | | | | | | | | 4: TxPDO | | | | | | | |
| | | | | | | | | 5: RxPDO | | | | | | | |
| | | | | | | | | 6: Remote transmission request of a TxPDO | | | | | | | |
| | | | | | | | | 7: Remote transmission request of a RxPDO | | | | | | | |
| | | | | | | | 8...15: reserved | | | | | | | | |
| Byte 8 | Control-Byte in the CAN telegram: | | | | | | | | | | | | | | |
| | write access: | | | | | | | 0x23=4Byte, 0x27=3Byte, 0x2B=2Byte, 0x2F-F=1Byte | | | | | | | |
| | read access: | | | | | | | 0x40 | | | | | | | |
| Byte 9 | Low Byte of the CAN object number (Index) | | | | | | | | | | | | | | |
| Byte 10 | High Byte of the CAN object number (Index) | | | | | | | | | | | | | | |
| Byte 11 | Subindex according to CANopen Specification for the drive | | | | | | | | | | | | | | |
| Byte 12 | Data with a write access (Low Byte) | | | | | | | | | | | | | | |
| Byte 13 | Data with a write access | | | | | | | | | | | | | | |
| Byte 14 | Data with a write access | | | | | | | | | | | | | | |

| | |
|---------|--------------------------------------|
| Byte 15 | Data with a write access (High Byte) |
|---------|--------------------------------------|

The drive answers every telegram with an answer in the Mailbox Input buffer.

20.13.2 Mailbox Input

The drive answers every CoE telegram with a 16 byte answer telegram in the Mailbox Input buffer. The bytes are defined as follows:

| Address 0x1C00 | | | | | | | | Address 0x1C0F | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| CAN over EtherCAT specific data (CoE Header) | | | | | | | | CAN specific data (standard CAN SDO) | | | | | | | |
| Byte 0 | Length of the data (Low Byte) | | | | | | | | | | | | | | |
| Byte 1 | Length of the data (High Byte) | | | | | | | | | | | | | | |
| Byte 2 | Address (Low Byte) | | | | | | | | | | | | | | |
| Byte 3 | Address (High Byte) | | | | | | | | | | | | | | |
| Byte 4 | Bit 0 to 5: Channel Bit 6 to 7: Priority | | | | | | | | | | | | | | |
| Byte 5 | Bit 0 to 3: Type | | | | | | | 1 = Reserved: ADS over EtherCAT 2 = Reserved: Ethernet over EtherCAT 3 = Can over EtherCAT... | | | | | | | |
| | Bit 4 to 7: Reserved | | | | | | | | | | | | | | |
| Byte 6 | PDO Number (with PDO transmissions only, Bit 0 = LSB of the PDO number, see Byte 7 for MSB) | | | | | | | | | | | | | | |
| Byte 7 | Bit 0: MSB of the PDO number, see Byte 6 | | | | | | | | | | | | | | |
| | Bit 1 to 3: Reserved | | | | | | | | | | | | | | |
| | Bit 4 to 7: CoE specific type | | | | | | | 0: Reserved | | | | | | | |
| | | | | | | | | 1: Emergency message | | | | | | | |
| | | | | | | | | 2: SDO request | | | | | | | |
| | | | | | | | | 3: SDO answer | | | | | | | |
| | | | | | | | | 4: TXPDO | | | | | | | |
| | | | | | | | | 5: RxPDO | | | | | | | |
| | | | | | | | | 6: Remote transmission request of a TxPDO | | | | | | | |
| | | | | | | | | 7: Remote transmission request of a RxPDO | | | | | | | |
| Byte 8 | Control-Byte in the CAN telegram: | | | | | | | | | | | | | | |
| | write access OK: | | | | | | | 0x60 | | | | | | | |
| | read access OK + length of answer: | | | | | | | 0x43 (4 Byte), 0x47 (3 Byte), 0x4B (2Byte), 0x4F (1Byte) | | | | | | | |
| | error with read- or write access: | | | | | | | 0x80 | | | | | | | |
| Byte 9 | Low Byte of the CAN object number (Index) | | | | | | | | | | | | | | |
| Byte 10 | High Byte of the CAN object number (Index) | | | | | | | | | | | | | | |
| Byte 11 | Subindex according to CANopen Specification for Kollmorgen drive | | | | | | | | | | | | | | |
| Byte 12 | Data (Low Byte) | | | | | | | | | | | | | | |

| | | |
|---------|------------------|--|
| Byte 13 | Data | error code Fehlercode according to CANopen Specification in case of an error |
| Byte 14 | Data | data value of the object in case of successful read access |
| Byte 15 | Data (High Byte) | |

20.13.3 Example: Mailbox Access

In the example below, PDOs 0x1704 are mapped (see Chapter "Fixed PDO Mappings" (→ p. 943) "Fixed PDO Mappings"):

The master sends this mailbox output message:

| | | |
|---------|------|--|
| Byte 0 | 0x0A | The next 10 Bytes contain data (Byte 2 to Byte 11) |
| Byte 1 | 0x00 | The next 10 Bytes contain data (Byte 2 to Byte 11) |
| Byte 2 | 0x00 | Address 0 |
| Byte 3 | 0x00 | Address 0 |
| Byte 4 | 0x00 | Channel 0 and Priority 0 |
| Byte 5 | 0x03 | CoE Object |
| Byte 6 | 0x00 | PDO Number 0 |
| Byte 7 | 0x20 | PDO Number 0 and SDO-Request |
| Byte 8 | 0x2B | 2 Byte write access |
| Byte 9 | 0x12 | SDO-Object 0x1C12 |
| Byte 10 | 0x1C | SDO-Object 0x1C12 |
| Byte 11 | 0x01 | Subindex 1 |
| Byte 12 | 0x04 | Data value 0x00001704 |
| Byte 13 | 0x17 | Data value 0x00001704 |
| Byte 14 | 0x00 | Data value 0x00001704 |
| Byte 15 | 0x00 | Data value 0x00001704 |

The drive returns the following message:

| | | |
|---------|------|--|
| Byte 0 | 0x0E | The next 14 Bytes contain data (Byte 2 to Byte 15) |
| Byte 1 | 0x00 | The next 14 Bytes contain data (Byte 2 to Byte 15) |
| Byte 2 | 0x00 | Address 0 |
| Byte 3 | 0x00 | Address 0 |
| Byte 4 | 0x00 | Channel 0 and Priority 0 |
| Byte 5 | 0x03 | CoE Object |
| Byte 6 | 0x00 | PDO Number 0 |
| Byte 7 | 0x20 | PDO Number 0 and SDO-Answer |
| Byte 8 | 0x60 | Successful write access |
| Byte 9 | 0x12 | SDO-Object 0x1C12 |
| Byte 10 | 0x1C | SDO-Object 0x1C12 |
| Byte 11 | 0x01 | Subindex 1 |
| Byte 12 | 0x00 | Data value 0x00000000 |
| Byte 13 | 0x00 | Data value 0x00000000 |
| Byte 14 | 0x00 | Data value 0x00000000 |
| Byte 15 | 0x00 | Data value 0x00000000 |

20.14 Important Configuration Parameters

The AKD PDMM holds several fieldbus-specific, general purpose parameters. Some of them contain the following relevant data:

FBUS.PARAM01:

Sets the baud rate for the CANbus. Supported baud rates are 125, 250, 500 and 1000 kBaud. On AKD PDMM-C, FBUS.PARAM01 sets and stores the EtherCAT station alias for the ESC (EtherCAT slave controller) of string 2.

FBUS.PARAM02:

This parameter activates the synchronization feature of the AKD PDMM. The DC feature must be activated in order to allow the AKD PDMM to get synchronized with the master. Only works when FBUS.TYPE = 3 (CANopen).

Drive internal PLL (phase locked loop) functionality: enabled (1),

Drive internal PLL functionality: disabled (0).

FBUS.PARAM03:

This parameter contains the Configured Station Alias address of the AKD PDMM. An EEPROM emulation write access to the Configured Station Alias address forces the AKD PDMM to store the drive parameters automatically using the DRV.NVSAVE command. On AKD PDMM-C, FBUS.PARAM03 sets and stores the EtherCAT station alias for the ESC (EtherCAT slave controller) of string 1.

FBUS.PARAM04:

This parameter enables (1) or disables (0) the synchronization supervision of the CANOpen or EtherCAT fieldbus.

Default values for this parameter are as follows:

CANopen drive: disabled (0)

EtherCAT drive: enabled (1)

Synchronization supervision is active when FBUS.PARAM 04 = 1 and the first CANOpen Sync message or first EtherCAT frame is received. When more than three CANOpen sync messages or seven EtherCAT frames have not been received and the drive is enabled, fault F125 ("Synchronization lost"), occurs.

FBUS.PARAM05:

| | | |
|-------|---|--|
| Bit 0 | 1 | Faults can only be reset using DS402 control word bit 7. |
| | 0 | The reset can also be done via telnet or digital input and the DS402 state machine reflects this condition. |
| Bit 1 | 1 | The state of the hardware enable does not change the state machine state Operation Enable. |
| | 0 | If the state Operation Enable or Switched on is active it falls back to the state switched On Disabled, if the Hardware enable goes to 0. |
| Bit 2 | 1 | EWV ¹ /Telnet can not software enable the drive, when CANopen/EtherCAT are Operational. |
| | 0 | EWV/Telnet can software enable the drive. NOTE: During commissioning this bit should be set to 1 to avoid influences to DS402 power stage state machine. The field bus should not be in operation as well to avoid influence to test functions of Workbench. |

¹Embedded Workbench Views

| | | |
|--|---|--|
| Bit 3 | 1 | DS402-state machine is not influenced, if the software-enable is taken away via Telnet. |
| | 0 | DS402-state machine is influenced, if the software-enable is taken away via Telnet. |
| Bit 4 | 1 | Scaling is done via special DS402 - objects (independent on units) |
| | 0 | Scaling for position, velocity and acceleration objects is done via UNIT parameters. |
| Bit 5 (EtherCAT only) | 1 | FBUS.PARAM03 defines the station alias address if not 0. If FBUS.PARAM03 set to 0, the address will be taken from rotary switches instead, if they are not 0. The EtherCAT master has the ability to use the alias address, selected by the drive, or issue its own. |
| | 0 | The rotary switches define the station alias address if not 0. If the rotary switches are set to 0, the address will be taken from FBUS.PARAM03 instead, if it is not 0. |
| Bit 6 | 1 | Bit 0 of parameter MT.CNTL (object 35B9 sub 0) can be accessed. |
| | 0 | Bit 0 of parameter MT.CNTL (object 35B9 sub 0) is exclusively used for DS402 controlword. |
| Bit 7 | 1 | All capture objects (0x20A0-0x20A3, 0x20A6, 0x20A7, 0x60BA to 0x60BD) are scaled like object 0x6063. |
| | 0 | All capture objects (0x20A0-0x20A3, 0x20A6, 0x20A7, 0x60BA to 0x60BD) are scaled like objectt 0x6064. |
| Bit 8 | 1 | DS402-state SWITCHED ON means power stage disabled. |
| | 0 | DS402-state SWITCHED ON means power stage enabled. |
| Bit 9 | 1 | SDO content of object 0x6063 is the same as PDO content. |
| | 0 | SDO content of object 0x6063 depends on AKD PDMM unit parameters. |
| Bit 10 (Bit 10 is active only, if Bit 8 is set) | 1 | State "Switch On" can be reached without the high-level voltage being active. |
| | 0 | State "Switch On" can only be reached when the high-level voltage is active; otherwise the drive will stay in "Ready to Switch On". |
| Bit 11 | 1 | No emergency messages over CANopen are triggered when a drive warning occurs. |
| | 0 | Emergency messages over CANopen are triggered when a drive warning occurs. |
| Bit 12 | | reserved |
| Bit 13 (EtherCAT only) | 1 | Downloaded parameter file is stored automatically to nonvolatile memory. |
| | 0 | Downloaded parameter file is not stored automatically to nonvolatile memory. |

FBUS.PARAM06 to FBUS.PARAM10:

Reserved.

20.15 EEPROM Content

AKD PDMM has a built-in emulated EEPROM. This EEPROM can be read by the EtherCAT master to get some information about drive properties, like PDO-information, drive name, serial numbers and communication-specific attributes.

They are organized in categories. There are two manufacturer-specific categories implemented in the AKD PDMM:

- Category 0x0800: Holds a string with the model type in the format AKD PDMM-P00000-NxxC-0000
- Category 0x0801: Holds the firmware version in the format 0x_xx-xx-yyy

This page intentionally left blank.

21 Appendix

21.1 CANopen Emergency Messages and Error Codes

Emergency messages are triggered by internal equipment errors. They have a high ID-priority to ensure quick access to the bus. An emergency message contains an error field with pre-defined error/fault numbers (2 bytes), an error register (1byte), the error category (1 byte), and additional information. Error numbers 0000h to 7FFFh are defined in the communication or drive profile. Error numbers FF00h to FFFFh have manufacturer-specific definitions.

| Error Code | Fault/Warning | Description |
|------------|---------------|--|
| 0x0000 | 0 | Emergency error free. |
| 0x1080 | - | General Warning. |
| 0x1081 | - | General Error. |
| 0x3110 | F523 | DC Bus link over voltage FPGA. |
| 0x3120 | F247 | DC Bus link voltage exceed allowed thresholds. |
| 0x3130 | F503 | DC Bus link capacitor overload. |
| 0x3180 | n503 | Warning: DC Bus link capacitor overload. |
| 0x3210 | F501 | DC Bus link over-voltage. |
| 0x3220 | F502 | DC Bus Link under-voltage. |
| 0x3280 | n502 | Warning: DC Bus Link under-voltage. |
| 0x3281 | n521 | Warning: Dynamic Braking I ² T. |
| 0x3282 | F519 | Regen short circuit. |
| 0x3283 | n501 | Warning: DC Bus link over-voltage. |
| 0x4210 | F234 | Excessive temperature, device (control board). |
| 0x4310 | F235 | Excessive temperature, drive (heat sink). |
| 0x4380 | F236 | Power temperature sensor 2 high. |
| 0x4381 | F237 | Power temperature sensor 3 high. |
| 0x4382 | F535 | Power board overtemperature. |
| 0x4390 | n234 | Warning: Control temperature sensor 1 high. |
| 0x4391 | n235 | Warning: Power temperature sensor 1 high. |
| 0x4392 | n236 | Warning: Power temperature sensor 2 high. |
| 0x4393 | n237 | Warning: Power temperature sensor 3 high. |
| 0x4394 | n240 | Warning: Control temperature sensor 1 low. |
| 0x4395 | n241 | Warning: Power temperature sensor 1 low. |
| 0x4396 | n242 | Warning: Power temperature sensor 2 low. |
| 0x4397 | n243 | Warning: Control temperature sensor 1 low. |
| 0x4398 | F240 | Control temperature sensor 1 low. |
| 0x4399 | F241 | Power temperature sensor 1 low. |
| 0x439A | F242 | Power temperature sensor 2 low. |

| Error Code | Fault/Warning | Description |
|------------|---------------|---|
| 0x439B | F243 | Power temperature sensor 3 low. |
| 0x5113 | F512 | 5V0 under voltage. |
| 0x5114 | F505 | 1V2 under voltage. |
| 0x5115 | F507 | 2V5 under voltage. |
| 0x5116 | F509 | 3V3 under voltage. |
| 0x5117 | F514 | +12V0 under voltage. |
| 0x5118 | F516 | -12V0 under voltage. |
| 0x5119 | F518 | Analog 3V3 under voltage. |
| 0x5180 | F504 | 1V2 over voltage. |
| 0x5181 | F506 | 2V5 over voltage. |
| 0x5182 | F508 | 3V3 over voltage. |
| 0x5183 | F510 | 5V0 over voltage. |
| 0x5184 | F513 | +12V0 over voltage. |
| 0x5185 | F515 | -12V0 over voltage. |
| 0x5186 | F517 | Analog 3V3 over voltage. |
| 0x5530 | F105 | Hardware memory, non-volatile memory stamp invalid. |
| 0x5580 | F106 | Hardware memory, non-volatile memory data. |
| 0x5590 | F204 | Control board EEPROM read failed. |
| 0x5591 | F205 | Control board EEPROM corrupted serial num stamp. |
| 0x5592 | F206 | Control board EEPROM corrupted serial num data. |
| 0x5593 | F207 | Control board EEPROM corrupted parameter stamp. |
| 0x5594 | F208 | Control board EEPROM corrupted parameter data. |
| 0x5595 | F219 | Control board EEPROM write failed. |
| 0x55A0 | F209 | Power board EEPROM read failed. |
| 0x55A1 | F210 | Power board EEPROM corrupted serial num stamp. |
| 0x55A2 | F212 | Power board EEPROM corrupted serial num data. |
| 0x55A3 | F213 | Power board EEPROM corrupted parameter stamp. |
| 0x55A4 | F214 | Power board EEPROM corrupted parameter data. |
| 0x55A5 | F230 | Power board EEPROM write failed. |
| 0x55A6 | F232 | Power board EEPROM invalid data. |
| 0x55B0 | F248 | Option board EEPROM corrupted. |
| 0x55B1 | F249 | Option board upstream checksum. |
| 0x55B2 | F250 | Option board upstream checksum. |
| 0x55B3 | F251 | Option board watchdog. |
| 0x55B8 | F252 | Firmware and option board FPGA types are not compatible. |
| 0x55B9 | F253 | Firmware and option board FPGA versions are not compatible. |
| 0x55C0 | F621 | Control Board CRC fault. |
| 0x55C1 | F623 | Power Board CRC fault. |

| Error Code | Fault/ Warning | Description |
|------------|----------------|---|
| 0x55C2 | F624 | Power Board Watchdog fault. |
| 0x55C3 | F625 | Power Board Communication fault. |
| 0x55C4 | F626 | Power Board FPGA not configured. |
| 0x55C5 | F627 | Control Board Watchdog fault. |
| 0x55C6 | n103 | Warning: Resident FPGA . |
| 0x55C7 | n104 | Warning: Operational FPGA . |
| 0x6380 | F532 | Drive motor parameters setup incomplete. |
| 0x7180 | F301 | Motor overheat. |
| 0x7182 | F305 | Motor Brake open circuit. |
| 0x7183 | F306 | Motor Brake short circuit. |
| 0x7184 | F307 | Motor Brake applied during enable state. |
| 0x7185 | F436 | EnDAT overheated. |
| 0x7186 | n301 | Warning: Motor overheated. |
| 0x7187 | F308 | Voltage exceeds motor rating. |
| 0x7188 | F560 | Regen near capacity, could not prevent over voltage. |
| 0x7189 | F312 | Brake released when it should be applied. |
| 0x7305 | F417 | Broken wire in primary feedback. |
| 0x7380 | F402 | Feedback 1 Analog signal amplitude default. |
| 0x7381 | F403 | Feedback 1 EnDat communication fault. |
| 0x7382 | F404 | Feedback 1 illegal hall state. |
| 0x7383 | F405 | Feedback 1 BiSS watchdog. |
| 0x7384 | F406 | Feedback 1 BiSS multi cycle. |
| 0x7385 | F407 | Feedback 1 BiSS sensor. |
| 0x7386 | F408 | Feedback 1 SFD configuration. |
| 0x7387 | F409 | Feedback 1 SFD UART overrun. |
| 0x7388 | F410 | Feedback 1 SFD UART frame. |
| 0x7389 | F412 | Feedback 1 SFD UART parity. |
| 0x738A | F413 | Feedback 1 SFD transfer timeout. |
| 0x738C | F415 | Feedback 1 SFD mult. corrupt position. |
| 0x738D | F416 | Feedback 1 SFD Transfer incomplete. |
| 0x738E | F418 | Feedback 1 power supply fault. |
| 0x738F | F401 | Feedback 1 failed to set feedback. |
| 0x7390 | n414 | Warning: SFD single corrupted position. |
| 0x7391 | F419 | Encoder init procedure failed. |
| 0x7392 | F534 | Failed to read motor parameters from feedback device. |
| 0x7393 | F421 | SFD position sensor fault. |
| 0x7394 | F463 | Tamagawa encoder: overheat. |
| 0x7395 | n451 | Warning: Tamagawa encoder battery. |

| Error Code | Fault/Warning | Description |
|------------|---------------|--|
| 0x7396 | n423 | Warning: Non volatile memory error, multiturn overflow. |
| 0x7398 | F135 | Homing is needed. |
| 0x7399 | F468 | FB2.Source not set. |
| 0x739A | F469 | FB1.ENCRESES is not power of two. |
| 0x739B | F423 | Non volatile memory error, multiturn overflow. |
| 0x739C | F467 | Hiperface DSL fault. |
| 0x739D | F452 | Multiturn overflow not supported with this feedback. |
| 0x739E | F465 | Excessive shock detected by feedback device. |
| 0x73A0 | F453 | Tamagawa encoder: communication timeout. |
| 0x73A1 | F454 | Tamagawa encoder: communication transfer incomplete. |
| 0x73A2 | F456 | Tamagawa encoder: communication CRC. |
| 0x73A3 | F457 | Tamagawa encoder: communication start timeout. |
| 0x73A4 | F458 | Tamagawa encoder: communication UART overrun. |
| 0x73A5 | F459 | Tamagawa encoder: communication UART framing. |
| 0x73A6 | F460 | Tamagawa encoder: over speed. |
| 0x73A7 | F461 | Tamagawa encoder: contouring error. |
| 0x73A8 | F462 | Tamagawa encoder: counting overflow. |
| 0x73A9 | F464 | Tamagawa encoder: multiturn error. |
| 0x73AA | F451 | Tamagawa encoder: battery. |
| 0x73B0 | F486 | Motor velocity exceeds emulated encoder maximum speed. |
| 0x73B8 | F420 | FB3 EnDat communication fault. |
| 0x73C5 | F482 | Commutation not initialized. |
| 0x73C6 | F483 | Motor U phase missing. |
| 0x73C7 | F484 | Motor V phase missing. |
| 0x73C8 | F485 | Motor W phase missing. |
| 0x73D0 | F487 | Wake and Shake. Validating positive movement failed. |
| 0x73D1 | F489 | Wake and Shake. Validating negative movement failed. |
| 0x73D2 | F490 | Wake and Shake. Validating commutation angle time out. |
| 0x73D3 | F491 | Wake and Shake. Validating commutation angle moved too far. |
| 0x73D4 | F492 | Wake and Shake. Validating commutation angle required more than MOTOR.ICONT. |
| 0x73D5 | F493 | Invalid commutation detected, motor accelerates in wrong direction. |
| 0x8130 | F129 | Life Guard Error or Heartbeat Error. |
| 0x8180 | n702 | Warning: Fieldbus communication lost. |
| 0x8280 | n601 | Warning: Modbus data rate is too high. |
| 0x8311 | F304 | Motor foldback. |
| 0x8331 | F524 | Drive foldback. |

| Error Code | Fault/Warning | Description |
|------------|---------------|---|
| 0x8380 | n524 | Warning: Drive foldback. |
| 0x8381 | n304 | Warning: Motor foldback. |
| 0x8382 | n309 | Warning: Motor I ² t load. |
| 0x8383 | n580 | Warning: Using derivate of position when using sensorless feedback type in position mode. |
| 0x8384 | n581 | Warning: Zero velocity when using induction sensorless feedback type in position mode. |
| 0x8480 | F302 | Over speed. |
| 0x8481 | F703 | Emergency timeout occurred while axis should disable. |
| 0x8482 | F480 | Fieldbus command velocity too high. |
| 0x8483 | F481 | Fieldbus command velocity too low. |
| 0x8582 | n107 | Warning: Positive software position limit is exceeded. |
| 0x8583 | n108 | Warning: Negative software position limit is exceeded. |
| 0x8611 | F439 | Following error (user). |
| 0x8685 | F138 | Instability during autotune. |
| 0x8686 | n151 | Warning: Not enough distance to move; Motion Exception. |
| 0x8687 | n152 | Warning: Not enough distance to move; Following Motion Exception. |
| 0x8688 | n153 | Warning: Velocity Limit Violation, Exceeding Max Limit. |
| 0x8689 | n154 | Warning: Following Motion Failed; Check Motion Parameters. |
| 0x868A | n156 | Warning: Target Position crossed due to Stop command. |
| 0x86A0 | n157 | Warning: Homing Index pulse not found. |
| 0x86A1 | n158 | Warning: Homing Reference Switch not found. |
| 0x86A2 | n159 | Warning: Failed to set motion task parameters. |
| 0x86A3 | n160 | Warning: Motion Task Activation Failed. |
| 0x86A4 | n161 | Warning: Homing Procedure Failed. |
| 0x86A5 | F139 | Target Position Over Short due to invalid Motion task activation. |
| 0x86A6 | n163 | Warning: MT.NUM exceeds limit. |
| 0x86A7 | n164 | Warning: Motion task is not initialized. |
| 0x86A8 | n165 | Warning: Motion task target position is out. |
| 0x86A9 | n167 | Warning: Software limit switch traversed. |
| 0x86AA | n168 | Warning: Invalid bit combination in the motion task control word. |
| 0x86AB | n169 | Warning: 1:1 profile cannot be triggered on the fly. |
| 0x86AC | n170 | Warning: Customer profile table is not initialized. |
| 0x86AD | n171 | Warning: Motion task activation is currently pending |
| 0x86AE | n135 | Warning: Homing is needed. |
| 0x86AF | n174 | Warning: Homing maximum distance exceeded |
| 0x86B0 | F438 | Following error (numeric). |
| 0x8780 | F125 | Fieldbus synchronization lost. |

| Error Code | Fault/Warning | Description |
|------------|---------------|---|
| 0x8781 | n125 | Warning: Fieldbus synchronization lost. |
| 0x8AF0 | n137 | Warning: Homing and feedback mismatch. |
| 0xFF00 | F701 | Fieldbus runtime. |
| 0xFF01 | F702 | Fieldbus communication lost. |
| 0xFF02 | F529 | Iu current offset limit exceeded. |
| 0xFF03 | F530 | Iv current offset limit exceeded. |
| 0xFF04 | F521 | Regen over power. |
| 0xFF07 | F525 | Output over current. |
| 0xFF08 | F526 | Current sensor short circuit. |
| 0xFF09 | F128 | MPOLES/FPOLES not an integer. |
| 0xFF0A | F531 | Power stage fault. |
| 0xFF0B | F602 | Safe torque off. |
| 0xFF0C | F131 | Secondary feedback A/B line break. |
| 0xFF0D | F130 | Secondary feedback supply over current. |
| 0xFF0E | F134 | Secondary feedback illegal state. |
| 0xFF0F | F245 | External fault. |
| 0xFF10 | F136 | Firmware and FPGA versions are not compatible. |
| 0xFF11 | F101 | Firmware type mismatch. |
| 0xFF12 | n439 | Warning: Following error (user). |
| 0xFF13 | n438 | Warning: Following error (numeric). |
| 0xFF14 | n102 | Warning: Operational FPGA is not a default FPGA. |
| 0xFF15 | n101 | Warning: The FPGA is a laboratory FPGA. |
| 0xFF16 | n602 | Warning: Safe torque off. |
| 0xFF17 | F132 | Secondary feedback Z line break. |
| 0xFF18 | n603 | Warning: OPMODE incompatible with CMDSOURCE. |
| 0xFF19 | n604 | Warning: EMUEMODE incompatible with DRV.HANDWHEELSRC. |

21.2 Object Dictionary

The following tables describe all objects reachable via SDO or PDO. (i.p. = in preparation).

Abbreviations:

| | | | |
|--------|------------------|-------|------------------|
| U | = UNSIGNED | RO | = Read only |
| INT | = INTEGER | RW | = Read and Write |
| VisStr | = Visible String | WO | = Write only |
| | | const | = Constant |

21.2.1 Float Scaling

The scaling applied to objects which match floating-point parameters in EWV¹/Telnet are listed in the column "Float Scaling."

For example, index 607Ah is listed as 1:1 - this means that commanding a value of 1000 in SDO 607Ah is equivalent to entering MT.P 1000.000 in EWV. On the other hand, index 3598h is listed as 1000:1 - this means that commanding a value of 1000 in SDO 3598h is equivalent to entering IL.KP 1.000 in EWV.

A few parameters are listed as variable (var), because the scaling depends on other settings.

21.2.2 Effectiveness of PDO set-points

Some objects are having effect only in the state machine state "Operation Enabled", which is controlled by the DS402 control word. They are marked with an asterisk (*) at the PDO mapping.

21.2.3 Communication SDOs

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|--------------|
| 1000h | 0 | U32 | | RO | no | Device type | — |
| 1001h | 0 | U8 | | RO | no | Error register | — |
| 1002h | 0 | U32 | | RO | yes | Manufacturer-specific status register | — |
| 1003h | | ARRAY | | | | Pre-defined error field | — |
| 1003h | 0 | U8 | | RW | no | Number of errors | — |
| 1003h | 1 to 10 | U32 | | RO | no | standard error field | — |
| 1005h | 0 | U32 | | RW | no | COB—ID SYNC message | — |
| 1006h | 0 | U32 | | RW | no | Communication cycle period | — |
| 1008h | 0 | VisStr | | const | no | Manufacturer device name | — |
| 1009h | 0 | VisStr | | const | no | Manufacturer hardware version | — |
| 100Ah | 0 | VisStr | | const | no | Manufacturer software version | — |
| 100Ch | 0 | U16 | | RW | no | Guard time | — |
| 100Dh | 0 | U8 | | RW | no | Lifetime factor | — |
| 1010h | | ARRAY | | | | Save parameters | — |
| 1010h | 0 | U8 | | RO | no | highest sub-index | — |
| 1010h | 1 | U32 | | RW | no | Saves the drive parameters from the RAM to the NV. | DRV.NVSAVE |
| 1011h | | ARRAY | | | | Load parameters | — |
| 1011h | 0 | U8 | | RO | no | highest sub-index | — |

¹Embedded Workbench Views

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--------------------------------------|--------------|
| 1011h | 1 | U32 | | RW | no | Loads default parameters to the RAM. | DRV.RSTVAR |
| 1012h | 0 | U32 | | RW | no | COB—ID for the Time Stamp | — |
| 1014h | 0 | U32 | | RW | no | COB—ID for the Emergency Object | — |
| 1016h | | RECORD | | | | Consumer heartbeat time | |
| 1016h | 0 | U8 | | RO | no | highest sub-index | — |
| 1016h | 1 | U32 | | RW | no | Consumer heartbeat time | — |
| 1017h | 0 | U16 | | RW | no | Producer heartbeat time | — |
| 1018h | | RECORD | | | | Identity Object | — |
| 1018h | 0 | U8 | | RO | no | highest sub-index | — |
| 1018h | 1 | U32 | | RO | no | Vendor ID | — |
| 1018h | 2 | U32 | | RO | no | Product Code | — |
| 1018h | 3 | U32 | | RO | no | Revision number | — |
| 1018h | 4 | U32 | | RO | no | Serial number | — |
| 1026h | | ARRAY | | | | OS prompt | — |
| 1026h | 0 | U8 | | RO | no | highest sub-index | — |
| 1026h | 1 | U8 | | WO | no | StdIn | — |
| 1026h | 2 | U8 | | RO | no | StdOut | — |
| 1400h | | RECORD | | | | RXPDO1 communication parameter | — |
| 1400h | 0 | U8 | | RO | no | highest sub-index | — |
| 1400h | 1 | U32 | | RW | no | RXPDO1 COB — ID | — |
| 1400h | 2 | U8 | | RW | no | Transmission type RXPDO1 | — |
| 1401h | | RECORD | | | | RXPDO2 communication parameter | — |
| 1401h | 0 | U8 | | RO | no | highest sub-index | — |
| 1401h | 1 | U32 | | RW | no | RXPDO2 COB—ID | — |
| 1401h | 2 | U8 | | RW | no | Transmission type RXPDO2 | — |
| 1402h | | RECORD | | | | RXPDO3 communication parameter | — |
| 1402h | 0 | U8 | | RO | no | highest sub-index | — |
| 1402h | 1 | U32 | | RW | no | RXPDO3 COB—ID | — |
| 1402h | 2 | U8 | | RW | no | Transmission type RXPDO3 | — |
| 1403h | | RECORD | | | | RXPDO4 communication parameter | — |
| 1403h | 0 | U8 | | RO | no | highest sub-index | — |
| 1403h | 1 | U32 | | RW | no | RXPDO4 COB—ID | — |
| 1403h | 2 | U8 | | RW | no | Transmission type RXPDO4 | — |
| 1600h | | RECORD | | | | RXPDO1 mapping parameter | — |
| 1600h | 0 | U8 | | RO | no | highest sub-index | — |
| 1600h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1601h | | RECORD | | | | RXPDO2 mapping parameter | — |
| 1601h | 0 | U8 | | RO | no | highest sub-index | — |
| 1601h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1602h | | RECORD | | | | RXPDO3 mapping parameter | — |
| 1602h | 0 | U8 | | RO | no | highest sub-index | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|-------------------------------------|--------------|
| 1602h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1603h | | RECORD | | | | RXPDO4 mapping parameter | — |
| 1603h | 0 | U8 | | RO | no | highest sub-index | — |
| 1603h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1800h | | RECORD | | | | TXPDO1 communication parameter | — |
| 1800h | 0 | U8 | | RO | no | highest sub-index | — |
| 1800h | 1 | U32 | | RW | no | TXPDO1 COB—ID | — |
| 1800h | 2 | U8 | | RW | no | Transmission type TXPDO1 | — |
| 1800h | 3 | U16 | | RW | no | Inhibit time | — |
| 1800h | 4 | U8 | | const | no | reserved | — |
| 1800h | 5 | U16 | | RW | no | Event timer | — |
| 1801h | | RECORD | | | | TXPDO2 communication parameter | — |
| 1801h | 0 | U8 | | RO | no | highest sub-index | — |
| 1801h | 1 | U32 | | RW | no | TXPDO2 COB—ID | — |
| 1801h | 2 | U8 | | RW | no | Transmission type TXPDO2 | — |
| 1801h | 3 | U16 | | RW | no | Inhibit time | — |
| 1801h | 4 | U8 | | const | no | reserved | — |
| 1801h | 5 | U16 | | RW | no | Event timer | — |
| 1802h | | RECORD | | | | TXPDO3 communication parameter | — |
| 1802h | 0 | U8 | | RO | no | highest sub-index | — |
| 1802h | 1 | U32 | | RW | no | TXPDO3 COB—ID | — |
| 1802h | 2 | U8 | | RW | no | Transmission type TXPDO3 | — |
| 1802h | 3 | U16 | | RW | no | Inhibit time | — |
| 1802h | 4 | U8 | | const | no | reserved | — |
| 1802h | 5 | U16 | | RW | no | Event timer | — |
| 1803h | | RECORD | | | | TXPDO4 communication parameter | — |
| 1803h | 0 | U8 | | RO | no | highest sub-index | — |
| 1803h | 1 | U32 | | RW | no | TXPDO4 COB—ID | — |
| 1803h | 2 | U8 | | RW | no | Transmission type TXPDO4 | — |
| 1803h | 3 | U16 | | RW | no | Inhibit time | — |
| 1803h | 4 | U8 | | const | no | reserved | — |
| 1803h | 5 | U16 | | RW | no | Event timer | — |
| 1A00h | | RECORD | | | | Mapping parameter TXPDO1 | — |
| 1A00h | 0 | U8 | | RO | no | highest sub-index | — |
| 1A00h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1A01h | | RECORD | | | | Mapping parameter TXPDO2 | — |
| 1A01h | 0 | U8 | | RO | no | highest sub-index | — |
| 1A01h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |
| 1A02h | | RECORD | | | | Mapping parameter TXPDO3 | — |
| 1A02h | 0 | U8 | | RO | no | highest sub-index | — |
| 1A02h | 1 to 8 | U32 | | RW | no | Mapping for n—th application object | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--------------------------------------|--------------|
| 1A03h | | RECORD | | | | Mapping parameter TXPDO4 | — |
| 1A03h | 0 | U8 | | RO | no | highest sub-index | — |
| 1A03h | 1 to 8 | U32 | | RW | no | Mapping for n—the application object | — |
| 1C12h | | ARRAY | | RW | no | RxPDO assign | — |
| 1C12h | 0 | U8 | | RO | no | highest sub-index | — |
| 1C13h | 1 to 4 | U8 | | RW | no | Subindex 001..004 | — |
| 1C13h | | ARRAY | | RW | no | TxPDO assign | — |
| 1C13h | 0 | U8 | | RO | no | highest sub-index | — |
| 1C13h | 1 to 4 | U8 | | RW | no | Subindex 001..004 | — |

21.2.4 Manufacturer specific SDOs

Objects 2000h to 3999h

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|---------------|
| 2000h | | ARRAY | | | | System Warnings | — |
| 2000h | 0 | U8 | | RO | no | highest sub-index | — |
| 2000h | 1 | U32 | | RO | no | System Warning 1 | DRV.WARNING1 |
| 2000h | 2 | U32 | | RO | no | System Warning 2 | DRV.WARNING2 |
| 2000h | 3 | U32 | | RO | no | System Warning 3 | DRV.WARNING3 |
| 2001h | | ARRAY | | | | System Faults | — |
| 2001h | 0 | U8 | | RO | no | highest sub-index | — |
| 2001h | 1 | U32 | | RO | no | System Fault 1 | DRV.FAULT1 |
| 2001h | 2 | U32 | | RO | no | System Fault 2 | DRV.FAULT2 |
| 2001h | 3 | U32 | | RO | no | System Fault 3 | DRV.FAULT3 |
| 2001h | 4 | U32 | | RO | no | System Fault 4 | DRV.FAULT4 |
| 2001h | 5 | U32 | | RO | no | System Fault 5 | DRV.FAULT5 |
| 2001h | 6 | U32 | | RO | no | System Fault 6 | DRV.FAULT6 |
| 2001h | 7 | U32 | | RO | no | System Fault 7 | DRV.FAULT7 |
| 2001h | 8 | U32 | | RO | no | System Fault 8 | DRV.FAULT8 |
| 2001h | 9 | U32 | | RO | no | System Fault 9 | DRV.FAULT9 |
| 2001h | A | U32 | | RO | no | System Fault 10 | DRV.FAULT10 |
| 2002h | | ARRAY | | | | Manufacturer status bytes | — |
| 2002h | 0 | U8 | | RO | no | highest sub-index | — |
| 2002h | 1 | U8 | | RO | yes | Manufacturer status bytes 1 | — |
| 2002h | 2 | U8 | | RO | yes | Manufacturer status bytes 2 | — |
| 2002h | 3 | U8 | | RO | yes | Manufacturer status bytes 3 | — |
| 2002h | 4 | U8 | | RO | yes | Manufacturer status bytes 4 | — |
| 2011h | | VAR | | RO | | DRV.RUNTIME in seconds | DRV.RUNTIME |
| 2012h | | ARRAY | | | | Fault history: Fault numbers | DRV.FAULTHIST |
| 2012h | 0 | U8 | | RO | no | highest sub-index | — |
| 2012h | 1 to 20 | U32 | | RO | no | Nth-latest entry in fault number list of fault history table | — |
| 2013h | | ARRAY | | | | Fault history: Time stamps | DRV.FAULTHIST |
| 2013h | 0 | U8 | | RO | no | highest sub-index | — |
| 2013h | 1 to 20 | U32 | | RO | no | Nth-latest entry in fault time stamp list of fault history table | — |
| 2014h | | ARRAY | | | | Mask TxPDO Channel 1 | — |
| 2014h | 1 | U32 | | RW | no | Mask (Byte 0..3) | — |
| 2014h | 2 | U32 | | RW | no | Mask (Byte 4..7) | — |
| 2015h | | ARRAY | | | | Mask TxPDO Channel 2 | — |
| 2015h | 1 | U32 | | RW | no | Mask (Byte 0..3) | — |
| 2015h | 2 | U32 | | RW | no | Mask (Byte 4..7) | — |
| 2016h | | ARRAY | | | | Mask TxPDO Channel 3 | — |
| 2016h | 1 | U32 | | RW | no | Mask (Byte 0..3) | — |
| 2016h | 2 | U32 | | RW | no | Mask (Byte 4..7) | — |
| 2017h | | ARRAY | | | | Mask TxPDO Channel 4 | — |
| 2017h | 1 | U32 | | RW | no | Mask (Byte 0..3) | — |
| 2017h | 2 | U32 | | RW | no | Mask (Byte 4..7) | — |
| 2018h | | ARRAY | | | | Firmware version | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|-----------------------|
| 2018h | 0 | U16 | | const | no | highest sub-index | — |
| 2018h | 1 | U16 | | const | no | Major version | — |
| 2018h | 2 | U16 | | const | no | Minor version | — |
| 2018h | 3 | U16 | | const | no | Revision | — |
| 2018h | 4 | U16 | | const | no | Branch version | — |
| 2026h | | ARRAY | | | | ASCII Channel | — |
| 2026h | 0 | U8 | | RO | no | highest sub-index | — |
| 2026h | 1 | VisStr | | WO | no | Command | — |
| 2026h | 2 | VisStr | | RO | no | Response | — |
| 2031h | 0 | VisStr | | RW | no | Drive Name, length 10 bits | DRV.NAME |
| 2032h | 0 | VisStr | | RW | no | Drive custom identifier string, length 32 byte | DRV.CUSTOM-IDENTIFIER |
| 204Ch | | ARRAY | | | | pv scaling factor | — |
| 204Ch | 0 | U8 | | RO | no | highest sub-index | — |
| 204Ch | 1 | INT32 | | RW | no | pv scaling factor numerator | — |
| 204Ch | 2 | INT32 | | RW | no | pv scaling factor denominator | — |
| 2050h | 0 | INT32 | 1:1 | RO | yes | Position, secondary feedback | DRV.HANDWHEEL |
| 2071h | 0 | INT32 | | RW | yes* | Target current | - |
| 2077h | 0 | INT32 | | RO | yes | Current actual value | - |
| 20A0h | 0 | INT32 | var | RO | yes | Latch position 1, positive edge | CAP0.PLFB , CAP0.T |
| 20A1h | 0 | INT32 | var | RO | yes | Latch position 1, negative edge | CAP0.PLFB , CAP0.T |
| 20A2h | 0 | INT32 | var | RO | yes | Latch position 2, positive edge | CAP1.PLFB , CAP1.T |
| 20A3h | 0 | INT32 | var | RO | yes | Latch position 2, negative edge | CAP1.PLFB , CAP1.T |
| 20A4h | 0 | U16 | | RW | yes | Latch control register | — |
| 20A5h | 0 | U16 | | RW | yes | Latch status register | — |
| 20A6h | 0 | INT32 | var | RO | yes | Gets captured position value | CAP0.PLFB |
| 20A7h | 0 | INT32 | var | RO | yes | Gets captured position value | CAP1.PLFB |
| 20B8h | 0 | U16 | | RW | yes | Clear changed digital input information | — |
| 3405h | | ARRAY | | | | VL.ARTYPE | — |
| 3405h | 0 | U8 | | RO | no | highest sub-index | — |
| 3405h | 1 | U8 | | RW | no | Calculation method for BiQuad filter 1 | VL.ARTYPE1 |
| 3405h | 2 | U8 | | RW | no | Calculation method for BiQuad filter 2 | VL.ARTYPE2 |
| 3405h | 3 | U8 | | RW | no | Calculation method for BiQuad filter 3 | VL.ARTYPE3 |
| 3405h | 4 | U8 | | RW | no | Calculation method for BiQuad filter 4 | VL.ARTYPE4 |
| 3406h | | ARRAY | | | | VL BiQuad | — |
| 3406h | 0 | U8 | | RO | no | highest sub-index | — |
| 3406h | 1 | U32 | 1000:1 | RW | no | Natural frequency of pole of anti-resonance (AR) filter 1 | VL.ARPF1 |
| 3406h | 2 | U32 | 1000:1 | RW | no | Natural frequency of pole of anti-resonance (AR) filter 2 | VL.ARPF2 |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|----------------------|
| 3406h | 3 | U32 | 1000:1 | RW | no | Natural frequency of pole of anti-resonance (AR) filter 3 | VL.ARPF3 |
| 3406h | 4 | U32 | 1000:1 | RW | no | Natural frequency of pole of anti-resonance (AR) filter 4 | VL.ARPF4 |
| 3406h | 5 | U32 | 1000:1 | RW | no | Q of pole of anti-resonance (AR) filter 1 | VL.ARPQ1 |
| 3406h | 6 | U32 | 1000:1 | RW | no | Q of pole of anti-resonance (AR) filter 2 | VL.ARPQ2 |
| 3406h | 7 | U32 | 1000:1 | RW | no | Q of pole of anti-resonance (AR) filter 3 | VL.ARPQ3 |
| 3406h | 8 | U32 | 1000:1 | RW | no | Q of pole of anti-resonance (AR) filter 4 | VL.ARPQ4 |
| 3406h | 9 | U32 | 1000:1 | RW | no | Natural frequency of zero of anti-resonance (AR) filter 1 | VL.ARZF1 |
| 3406h | A | U32 | 1000:1 | RW | no | Natural frequency of zero of anti-resonance (AR) filter 2 | VL.ARZF2 |
| 3406h | B | U32 | 1000:1 | RW | no | Natural frequency of zero of anti-resonance (AR) filter 3 | VL.ARZF3 |
| 3406h | C | U32 | 1000:1 | RW | no | Natural frequency of zero of anti-resonance (AR) filter 4 | VL.ARZF4 |
| 3406h | D | U32 | 1000:1 | RW | no | Q of zero of anti-resonance filter 1 | VL.ARZQ1 |
| 3406h | E | U32 | 1000:1 | RW | no | Q of zero of anti-resonance filter 2 | VL.ARZQ2 |
| 3406h | F | U32 | 1000:1 | RW | no | Q of zero of anti-resonance filter 3 | VL.ARZQ3 |
| 3406h | 10 | U32 | 1000:1 | RW | no | Q of zero of anti-resonance filter 4 | VL.ARZQ4 |
| 3407h | | STRUCT | | | | Velocity Filter | — |
| 3407h | 0 | U8 | | RO | no | highest sub-index | — |
| 3407h | 1 | INT32 | 1000:1 | RW | no | 10 Hz filtered VL.FB | VL.FBFILTER |
| 3407h | 2 | U32 | 1000:1 | RW | no | Gain for the velocity feed-forward | VL.KVFF |
| 3407h | 3 | U32 | | RW | no | Gain for the acceleration feed-forward | VL.KBUSFF |
| 3407h | 4 | U32 | 1:1 | RW | no | Sets the velocity error | VL.ERR |
| 3412h | 0 | INT8 | | RW | no | Type of regen resistor | REGEN.TYPE |
| 3414h | 0 | U8 | | RW | | Returns and sets the regen resistor fault level temperature. | REGEN.WATTEXT |
| 3415h | 0 | U32 | 1000:1 | RO | no | Thermal regen resistor time constant | REGEN.TEXT |
| 3416h | 0 | U32 | | RO | no | Gets regen resistor's calculated power | REGEN.POWER |
| 3417h | 0 | U32 | | RO | no | Returns a filtered version of 3416h | REGEN.POWER-FILTERED |
| 3420h | 0 | U16 | 1000:1 | RW | no | Sets the foldback fault level. | IL.FOLDFTHRESH |
| 3421h | 0 | U32 | 1000:1 | RW | no | Sets the user value for the foldback fault level. | IL.FOLDFTHRESHU |
| 3422h | 0 | U32 | 1000:1 | | no | Sets friction compensation value. | IL.FRICTION |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|-------------------------------|
| 3423h | 0 | INT32 | 1000:1 | | no | A constant current command added to compensate for gravity. | IL.OFFSET |
| 3424h | 0 | U16 | | | no | Enables/disables the integrator part of the PI loop. | IL.INTEN (Password Protected) |
| 3425h | 0 | U32 | 1000:1 | RO | no | Reads the overall foldback current limit | IL.IFOLD |
| 3426h | 0 | U32 | 1000:1 | RW | no | Sets current loop acceleration feedforward gain value | IL.KACFF |
| 3427h | | RECORD | | | | Motor protection parameters | — |
| 3427h | 0 | U8 | | RO | no | highest sub-index | — |
| 3427h | 1 | U8 | | RW | no | | IL.MIMODE |
| 3427h | 2 | U8 | | RW | no | | IL.MI2TWTRESH |
| 3427h | 3 | U32 | | RW | yes | | IL.MI2T |
| 3430h | 0 | U8 | | RW | no | Sets the direction for absolute motion tasks. | PL.MODPDIR |
| 3431h | 0 | U16 | | RW | no | Sets the motion task in the drive | MT.SET |
| 3440h | | ARRAY | | | | Controlled stop parameters | — |
| 3440h | 0 | U8 | | RO | no | highest sub-index | — |
| 3440h | 1 | U32 | 1:1 | RW | no | Sets the deceleration value for a controlled stop. | CS.DEC |
| 3440h | 2 | U32 | 1:1 | RW | no | Sets the velocity threshold for a controlled stop. | CS.VTHRESH |
| 3440h | 3 | U32 | | RW | no | Sets the time value for the drive velocity to be within CS.VTHRESH. | CS.TO |
| 3441h | 0 | U8 | | RO | no | Controlled stop state | CS.STATE |
| 3443h | 0 | U16 | | RO | no | Returns the possible reason for a drive disable | DRV.DIS |
| 3444h | 0 | U16 | 1000:1 | RO | no | Maximum current for dynamic braking | DRV.DBILIMIT |
| 3445h | 0 | U32 | | RO | no | Emergency timeout for braking | DRV.DISTO |
| 3450h | 0 | U8 | | WO | no | Release or enable brake | MOTOR.BRAKERLS |
| 3451h | 0 | U8 | | RW | no | Determines which drive parameters are calculated automatically. | MOTOR.AUTOSSET |
| 3452h | 0 | U16 | | RW | no | Sets the motor maximum voltage | MOTOR.VOLTMAX |
| 3453h | 0 | U32 | | RW | no | Sets the motor temperature warning level | MOTOR.TEMPWARN |
| 3454h | 0 | U32 | 1000:1 | RW | no | Sets the thermal constant of the motor coil | MOTOR.CTF0 |
| 3455h | 0 | U32 | 1000:1 | RW | no | Sets the line-to-line motor Lq | MOTOR.LQLL |
| 3456h | 0 | U32 | 1000:1 | RW | no | Sets the stator winding resistance phase-phase in ohms | MOTOR.R |
| 3457h | | RECORD | | | | Induction Motor parameter | — |
| 3457h | 0 | U8 | | RO | no | highest sub-index | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|-----------------|
| 3457h | 1 | INT32 | 1000:1 | RW | no | Configuration of induction motor's rated velocity. | MOTOR.VRATED |
| 3457h | 2 | U16 | | RW | no | Configuration of induction motor's rated voltage. | MOTOR.VOLTRATED |
| 3457h | 3 | U16 | | RW | no | Sets the minimum voltage for V/f Control. | MOTOR.VOLTMIN |
| 345Ah | | ARRAY | | | | Brake Control | — |
| 345Ah | 0 | U8 | | RO | no | highest sub-index | — |
| 345Ah | 1 | U16 | | RW | yes | Brake Control Command | — |
| 345Ah | 2 | U16 | | RO | yes | Brake Status Response. | — |
| 3460h | | RECORD | | | | Capture engines parameters | — |
| 3460h | 0 | U8 | | RO | no | highest sub-index | — |
| 3460h | 1 | U8 | | RW | no | Specifies the trigger source for the position capture. | CAP0.TRIGGER |
| 3460h | 2 | U8 | | RW | no | Specifies the trigger source for the position capture. | CAP1.TRIGGER |
| 3460h | 3 | U8 | | RW | no | Selects the captured value. | CAP0.MODE |
| 3460h | 4 | U8 | | RW | no | Selects the captured value. | CAP1.MODE |
| 3460h | 5 | U8 | | RW | no | Controls the precondition logic. | CAP0.EVENT |
| 3460h | 6 | U8 | | RW | no | Controls the precondition logic. | CAP1.EVENT |
| 3460h | 7 | U8 | | RW | no | Selects the capture precondition edge. | CAP0.PREEDGE |
| 3460h | 8 | U8 | | RW | no | Selects the capture precondition edge. | CAP1.PREEDGE |
| 3460h | 9 | U8 | | RW | no | Sets the precondition trigger. | CAP0.PRESELECT |
| 3460h | A | U8 | | RW | no | Sets the precondition trigger. | CAP1.PRESELECT |
| 3460h | B | U8 | | RW | no | Selects the feedback source for the capture engine 0. | CAP0.FBSOURCE |
| 3460h | C | U8 | | RW | no | Selects the feedback source for the capture engine 1. | CAP1.FBSOURCE |
| 3470h | | RECORD | | | | | — |
| 3470h | 0 | U8 | | RO | no | highest sub-index | — |
| 3470h | 1 | INT8 | | RW | no | Sets the analog output mode. | AOUT.MODE |
| 3470h | 2 | INT16 | 1000:1 | RW | yes | Reads the analog output value. | AOUT.VALUE |
| 3470h | 3 | INT16 | 1000:1 | RW | yes | Reads and writes the analog output value. | AOUT.VALUEU |
| 3470h | 4 | INT16 | 1000:1 | RO | yes | Reads the value of the analog input signal. | AIN.VALUE |
| 3470h | 5 | U32 | 1000:1 | RW | no | Sets velocity scale factor for analog output | AOUT.VSCALE |
| 3471h | 0 | U32 | 1:1 | RW | no | Sets the analog position scale factor | AOUT.PSCALE |
| 3472h | 0 | U32 | 1:1 | RW | no | Sets analog pscale factor | AIN.PSCALE |
| 3474h | | ARRAY | | | | DINx.PARAM | — |
| 3474h | 0 | U8 | | RO | no | highest sub-index | — |
| 3474h | 1 | U32 | | RW | no | Lower 32-bit part of input parameter 1 | DIN1.PARAM |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|--------------|
| 3474h | 2 | U32 | | RW | no | Lower 32-bit part of input parameter 2 | DIN2.PARAM |
| 3474h | 3 | U32 | | RW | no | Lower 32-bit part of input parameter 3 | DIN3.PARAM |
| 3474h | 4 | U32 | | RW | no | Lower 32-bit part of input parameter 4 | DIN4.PARAM |
| 3474h | 5 | U32 | | RW | no | Lower 32-bit part of input parameter 5 | DIN5.PARAM |
| 3474h | 6 | U32 | | RW | no | Lower 32-bit part of input parameter 6 | DIN6.PARAM |
| 3474h | 7 | U32 | | RW | no | Lower 32-bit part of input parameter 7 | DIN7.PARAM |
| 3474h | 8 | U32 | | RW | no | Higher 32-bit part of input parameter 1 | DIN1.PARAM |
| 3474h | 9 | U32 | | RW | no | Higher 32-bit part of input parameter 2 | DIN2.PARAM |
| 3474h | A | U32 | | RW | no | Higher 32-bit part of input parameter 3 | DIN3.PARAM |
| 3474h | 4 | U32 | | RW | no | Higher 32-bit part of input parameter 1 | DIN1.PARAM |
| 3474h | 5 | U32 | | RW | no | Higher 32-bit part of input parameter 2 | DIN2.PARAM |
| 3474h | 6 | U32 | | RW | no | Higher 32-bit part of input parameter 3 | DIN3.PARAM |
| 3474h | B | U32 | | RW | no | Higher 32-bit part of input parameter 4 | DIN4.PARAM |
| 3474h | C | U32 | | RW | no | Higher 32-bit part of input parameter 5 | DIN5.PARAM |
| 3474h | D | U32 | | RW | no | Higher 32-bit part of input parameter 6 | DIN6.PARAM |
| 3474h | E | U32 | | RW | no | Higher 32-bit part of input parameter 7 | DIN7.PARAM |
| 3475h | | ARRAY | | | | DOUTx.PARAM | — |
| 3475h | 0 | U8 | | RO | no | highest sub-index | — |
| 3475h | 1 | U32 | | RW | no | Lower 32-bit part of output parameter 1 | DOUT1.PARAM |
| 3475h | 2 | U32 | | RW | no | Lower 32-bit part of output parameter 2 | DOUT2.PARAM |
| 3475h | 3 | U32 | | RW | no | Higher 32-bit part of output parameter 1 | DOUT1.PARAM |
| 3475h | 2 | U32 | | RW | no | Higher 32-bit part of output parameter 1 | DOUT1.PARAM |
| 3475h | 4 | U32 | | RW | no | Higher 32-bit part of output parameter 2 | DOUT2.PARAM |
| 3480h | 0 | U32 | 1000:1 | RW | no | Integral gain of position regulator PID loop | PL.KI |
| 3481h | | ARRAY | | | | PL.INTMAX | — |
| 3481h | 0 | U8 | | RO | no | highest sub-index | — |
| 3481h | 1 | U32 | 1:1 | RW | no | Input saturation | PL.INTINMAX |
| 3481h | 2 | U32 | 1:1 | RW | no | Output saturation | PL.INTOUTMAX |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|-----------------|
| 3482h | 0 | INT32 | 1:1 | RO | no | Maximum value of following error in homing | HOME.PERRTHRESH |
| 3483h | 0 | INT32 | 1:1 | RW | no | Sets the position error warning level | PL.ERRWTHRESH |
| 3484h | 0 | INT32 | 1:1 | RW | no | Specification of an additional movement after homing is completed. | HOME.DIST |
| 3490h | 0 | INT32 | 1:1 | RW | no | Position feedback offset | FB1.OFFSET |
| 3491h | 0 | U32 | | RO | no | Location of index pulse on EEO | DRV.EMUETURN |
| 3492h | 0 | U32 | | RO | no | Motion status of the drive | DRV.MOTIONSTAT |
| 3493h | 0 | U8 | | RO | no | Direction of EEO (emulated encoder output) | DRV.EMUEDIR |
| 3494h | | RECORD | | | | WS parameters | — |
| 3494h | 0 | U8 | | RO | no | highest sub-index | — |
| 3494h | 1 | INT16 | 1000:1 | RW | no | Sets maximum current used for wake and shake | WS.IMAX |
| 3494h | 2 | INT32 | 1:1 | RW | no | Sets the maximum movement required for wake and shake | WS.DISTMAX |
| 3494h | 3 | U16 | | RW | no | Sets the delay for wake and shake between loops in mode 0 | WS.TDELAY30 |
| 3494h | 4 | INT32 | 1:1 | RW | no | Defines the maximum allowed velocity for Wake & Shake | WS.VTHRESH |
| 3494h | 5 | U8 | | RO | no | Reads wake and shake status | WS.STATE |
| 3494h | 6 | U8 | | RW | no | Arm Wake and Shake to start | WS.ARM |
| 3495h | 0 | U16 | 1000:1 | RW | no | Voltage level for undervoltage warning. | VBUS.UVWTHRESH |
| 3496h | | ARRAY | | | | FBUS synchronization parameters | — |
| 3496h | 0 | U8 | | RO | no | highest sub-index | — |
| 3496h | 1 | U32 | | RW | no | expected time distance between clearing the PLL counter and calling the PLL function | FBUS.SYNCDIST |
| 3496h | 2 | U32 | | RW | no | actual time distance between clearing the PLL counter and calling the PLL function | FBUS.SYNCACT |
| 3496h | 3 | U32 | | RW | no | Time window, which is used in order to consider the drive as being synchronized | FBUS.SYNCWND |
| 3496h | 4 | U32 | | RW | no | Time, which is used for extending or lowering the sample rate of the internal 16 [kHz] IRQ | — |
| 3498h | 0 | U8 | | RW | no | Protection level of fieldbus against other communication channels (Telnet, Modbus..) | FBUS.PROTECTION |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|-----------------|
| 3499h | 0 | INT32 | | RW | yes | Set-point for stepper motor output through the emulated encoder output (EEO) | DRV.EMUSTEPCMD |
| 34A0h | | ARRAY | | | | PLS Position | |
| 34A0h | 0 | U8 | | RO | no | highest sub-index | — |
| 34A0h | 1 | INT32 | 1:1 | RW | no | Limit switch 1 compare value | PLS.P1 |
| 34A0h | 2 | INT32 | 1:1 | RW | no | Limit switch 2 compare value | PLS.P2 |
| 34A0h | 3 | INT32 | 1:1 | RW | no | Limit switch 3 compare value | PLS.P3 |
| 34A0h | 4 | INT32 | 1:1 | RW | no | Limit switch 4 compare value | PLS.P4 |
| 34A0h | 5 | INT32 | 1:1 | RW | no | Limit switch 5 compare value | PLS.P5 |
| 34A0h | 6 | INT32 | 1:1 | RW | no | Limit switch 6 compare value | PLS.P6 |
| 34A0h | 7 | INT32 | 1:1 | RW | no | Limit switch 7 compare value | PLS.P7 |
| 34A0h | 8 | INT32 | 1:1 | RW | no | Limit switch 8 compare value | PLS.P8 |
| 34A1h | | ARRAY | | | | PLS Width | — |
| 34A1h | 0 | U8 | | RO | no | highest sub-index | — |
| 34A1h | 1 | INT32 | 1:1 | RW | no | Sets Limit Switch1 Width | PLS.WIDTH1 |
| 34A1h | 2 | INT32 | 1:1 | RW | no | Sets Limit Switch 2 Width | PLS.WIDTH2 |
| 34A1h | 3 | INT32 | 1:1 | RW | no | Sets Limit Switch 3 Width | PLS.WIDTH3 |
| 34A1h | 4 | INT32 | 1:1 | RW | no | Sets Limit Switch 4 Width | PLS.WIDTH4 |
| 34A1h | 5 | INT32 | 1:1 | RW | no | Sets Limit Switch 5 Width | PLS.WIDTH5 |
| 34A1h | 6 | INT32 | 1:1 | RW | no | Sets Limit Switch 6 Width | PLS.WIDTH6 |
| 34A1h | 7 | INT32 | 1:1 | RW | no | Sets Limit Switch 7 Width | PLS.WIDTH7 |
| 34A1h | 8 | INT32 | 1:1 | RW | no | Sets Limit Switch 8 Width | PLS.WIDTH8 |
| 34A2h | | ARRAY | | | | PLS Time | — |
| 34A2h | 0 | U8 | | RO | no | highest sub-index | — |
| 34A2h | 1 | U16 | | RW | no | Sets limit switch 1 time | PLS.T1 |
| 34A2h | 2 | U16 | | RW | no | Sets limit switch 2 time | PLS.T2 |
| 34A2h | 3 | U16 | | RW | no | Sets limit switch 3 time | PLS.T3 |
| 34A2h | 4 | U16 | | RW | no | Sets limit switch 4 time | PLS.T4 |
| 34A2h | 5 | U16 | | RW | no | Sets limit switch 5 time | PLS.T5 |
| 34A2h | 6 | U16 | | RW | no | Sets limit switch 6 time | PLS.T6 |
| 34A2h | 7 | U16 | | RW | no | Sets limit switch 7 time | PLS.T7 |
| 34A2h | 8 | U16 | | RW | no | Sets limit switch 8 time | PLS.T8 |
| 34A3h | | ARRAY | | | | PLS Configuration | — |
| 34A3h | 0 | U8 | | RO | no | highest sub-index | — |
| 34A3h | 1 | U16 | | RW | no | Enables the limit switches | PLS.EN |
| 34A3h | 2 | U16 | | RW | no | Resets limit switches | PLS.RESET |
| 34A3h | 3 | U16 | | RW | no | Selects limit switch mode | PLS.MODE |
| 34A3h | 4 | U16 | | RW | no | Reads the limit switch state | PLS.STATE |
| 34A4h | 0 | U8 | | RW | no | Sets limit switch units | PLS.UNITS |
| 34A8h | 0 | INT32 | | RW | no | Sets the Compare 0 modulo value | CMP0.MODVALUE |
| 34A9h | | ARRAY | | | | Compare0 modulo bounds | — |
| 34A9h | 0 | U8 | | RO | no | highest sub-index | — |
| 34A9h | 1 | U8 | | RW | no | Compare0 modulo bound 1 | CMP0.MODBOUND1 |
| 34A9h | 2 | U8 | | RW | no | Compare0 modulo bound 2 | CMP0.MODBOUND2 |
| 34AAh | | ARRAY | | | | CMP0 setpoints | — |
| 34AAh | 0 | U8 | | RO | no | highest sub-index | — |
| 34AAh | 1 | INT32 | | RW | no | Compare0 setpoint 0 | CMP0.SETPOINT 0 |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|------------------|
| 34AAh | 2 | INT32 | | RW | no | Compare0 setpoint 1 | CMP0.SETPOINT 1 |
| 34AAh | 3 | INT32 | | RW | no | Compare0 setpoint 2 | CMP0.SETPOINT 2 |
| 34AAh | 4 | INT32 | | RW | no | Compare0 setpoint 3 | CMP0.SETPOINT 3 |
| 34AAh | 5 | INT32 | | RW | no | Compare0 setpoint 4 | CMP0.SETPOINT 4 |
| 34AAh | 6 | INT32 | | RW | no | Compare0 setpoint 5 | CMP0.SETPOINT 5 |
| 34AAh | 7 | INT32 | | RW | no | Compare0 setpoint 6 | CMP0.SETPOINT 6 |
| 34AAh | 8 | INT32 | | RW | no | Compare0 setpoint 7 | CMP0.SETPOINT 7 |
| 34ABh | | ARRAY | | | | CMP0 widths | — |
| 34ABh | 0 | U8 | | RO | no | highest sub-index | — |
| 34ABh | 1 | INT32 | | RW | no | Compare0 width 0 | CMP0.WIDTH 0 |
| 34ABh | 2 | INT32 | | RW | no | Compare0 width 1 | CMP0.WIDTH 1 |
| 34ABh | 3 | INT32 | | RW | no | Compare0 width 2 | CMP0.WIDTH 2 |
| 34ABh | 4 | INT32 | | RW | no | Compare0 width 3 | CMP0.WIDTH 3 |
| 34ABh | 5 | INT32 | | RW | no | Compare0 width 4 | CMP0.WIDTH 4 |
| 34ABh | 6 | INT32 | | RW | no | Compare0 width 5 | CMP0.WIDTH 5 |
| 34ABh | 7 | INT32 | | RW | no | Compare0 width 6 | CMP0.WIDTH 6 |
| 34ABh | 8 | INT32 | | RW | no | Compare0 width 7 | CMP0.WIDTH 7 |
| 34ACh | | ARRAY | | | | CMP0 widthtype | — |
| 34ACh | 0 | U8 | | RO | no | highest sub-index | — |
| 34ACh | 1 | U8 | | RW | no | Compare0 widthtype 0 | CMP0.WIDTHTYPE 0 |
| 34ACh | 2 | U8 | | RW | no | Compare0 widthtype 1 | CMP0.WIDTHTYPE 1 |
| 34ACh | 3 | U8 | | RW | no | Compare0 widthtype 2 | CMP0.WIDTHTYPE 2 |
| 34ACh | 4 | U8 | | RW | no | Compare0 widthtype 3 | CMP0.WIDTHTYPE 3 |
| 34ACh | 5 | U8 | | RW | no | Compare0 widthtype 4 | CMP0.WIDTHTYPE 4 |
| 34ACh | 6 | U8 | | RW | no | Compare0 widthtype 5 | CMP0.WIDTHTYPE 5 |
| 34ACh | 7 | U8 | | RW | no | Compare0 widthtype 6 | CMP0.WIDTHTYPE 6 |
| 34ACh | 8 | U8 | | RW | no | Compare0 widthtype 7 | CMP0.WIDTHTYPE 7 |
| 34ADh | | ARRAY | | | | CMP0 modes | — |
| 34ADh | 0 | U8 | | RO | no | highest sub-index | — |
| 34ADh | 1 | U8 | | RW | no | Compare0 mode 0 | CMP0.MODE 0 |
| 34ADh | 2 | U8 | | RW | no | Compare0 mode 1 | CMP0.MODE 1 |
| 34ADh | 3 | U8 | | RW | no | Compare0 mode 2 | CMP0.MODE 2 |
| 34ADh | 4 | U8 | | RW | no | Compare0 mode 3 | CMP0.MODE 3 |
| 34ADh | 5 | U8 | | RW | no | Compare0 mode 4 | CMP0.MODE 4 |
| 34ADh | 6 | U8 | | RW | no | Compare0 mode 5 | CMP0.MODE 5 |
| 34ADh | 7 | U8 | | RW | no | Compare0 mode 6 | CMP0.MODE 6 |
| 34ADh | 8 | U8 | | RW | no | Compare0 mode 7 | CMP0.MODE 7 |
| 34B0h | | ARRAY | | | | USER.DWORDS for writing of feedback memory | — |
| 34B0h | 0 | U8 | | RO | no | highest sub-index | — |
| 34B0h | 1 | U32 | | RW | no | FB1.USERDWORD1 | FB1.USERDWORD1 |
| 34B0h | 2 | U32 | | RW | no | FB1.USERDWORD2 | FB1.USERDWORD2 |
| 34B1h | | ARRAY | | | | USER.WORDS for writing of feedback memory | — |
| 34B1h | 0 | U8 | | RO | no | highest sub-index | — |
| 34B1h | 1 | U16 | | RW | no | FB1.USERWORD1 | FB1.USERWORD1 |
| 34B1h | 2 | U16 | | RW | no | FB1.USERWORD2 | FB1.USERWORD2 |
| 34B1h | 3 | U16 | | RW | no | FB1.USERWORD3 | FB1.USERWORD3 |
| 34B1h | 4 | U16 | | RW | no | FB1.USERWORD4 | FB1.USERWORD4 |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|------------------|
| 34B2h | | ARRAY | | | | USER.BYTES for writing of feedback memory | — |
| 34B2h | 0 | U8 | | RO | no | highest sub-index | — |
| 34B2h | 1 | U8 | | RW | no | FB1.USERBYTE1 | FB1.USERBYTE1 |
| 34B2h | 2 | U8 | | RW | no | FB1.USERBYTE2 | FB1.USERBYTE2 |
| 34B2h | 3 | U8 | | RW | no | FB1.USERBYTE3 | FB1.USERBYTE3 |
| 34B2h | 4 | U8 | | RW | no | FB1.USERBYTE4 | FB1.USERBYTE4 |
| 34B2h | 5 | U8 | | RW | no | FB1.USERBYTE5 | FB1.USERBYTE5 |
| 34B2h | 6 | U8 | | RW | no | FB1.USERBYTE6 | FB1.USERBYTE6 |
| 34B2h | 7 | U8 | | RW | no | FB1.USERBYTE7 | FB1.USERBYTE7 |
| 34B2h | 8 | U8 | | RW | no | FB1.USERBYTE8 | FB1.USERBYTE8 |
| 34B8h | 0 | INT32 | | RW | no | Sets the Compare 1 modulo value | CMP1.MODVALUE |
| 34B9h | | ARRAY | | | | Compare1 modulo bounds | — |
| 34B9h | 0 | U8 | | RO | no | highest sub-index | — |
| 34B9h | 1 | U8 | | RW | no | Compare1 modulo bound 1 | CMP1.MODBOUND1 |
| 34B9h | 2 | U8 | | RW | no | Compare1 modulo bound 2 | CMP1.MODBOUND2 |
| 34BAh | | ARRAY | | | | CMP1 setpoints | — |
| 34BAh | 0 | U8 | | RO | no | highest sub-index | — |
| 34BAh | 1 | INT32 | | RW | no | Compare1 setpoint 0 | CMP1.SETPOINT 0 |
| 34BAh | 2 | INT32 | | RW | no | Compare1 setpoint 1 | CMP1.SETPOINT 1 |
| 34BAh | 3 | INT32 | | RW | no | Compare1 setpoint 2 | CMP1.SETPOINT 2 |
| 34BAh | 4 | INT32 | | RW | no | Compare1 setpoint 3 | CMP1.SETPOINT 3 |
| 34BAh | 5 | INT32 | | RW | no | Compare1 setpoint 4 | CMP1.SETPOINT 4 |
| 34BAh | 6 | INT32 | | RW | no | Compare1 setpoint 5 | CMP1.SETPOINT 5 |
| 34BAh | 7 | INT32 | | RW | no | Compare1 setpoint 6 | CMP1.SETPOINT 6 |
| 34BAh | 8 | INT32 | | RW | no | Compare1 setpoint 7 | CMP1.SETPOINT 7 |
| 34BBh | | ARRAY | | | | CMP1 widths | — |
| 34BBh | 0 | U8 | | RO | no | highest sub-index | — |
| 34BBh | 1 | INT32 | | RW | no | Compare1 width 0 | CMP1.WIDTH 0 |
| 34BBh | 2 | INT32 | | RW | no | Compare1 width 1 | CMP1.WIDTH 1 |
| 34BBh | 3 | INT32 | | RW | no | Compare1 width 2 | CMP1.WIDTH 2 |
| 34BBh | 4 | INT32 | | RW | no | Compare1 width 3 | CMP1.WIDTH 3 |
| 34BBh | 5 | INT32 | | RW | no | Compare1 width 4 | CMP1.WIDTH 4 |
| 34BBh | 6 | INT32 | | RW | no | Compare1 width 5 | CMP1.WIDTH 5 |
| 34BBh | 7 | INT32 | | RW | no | Compare1 width 6 | CMP1.WIDTH 6 |
| 34BBh | 8 | INT32 | | RW | no | Compare1 width 7 | CMP1.WIDTH 7 |
| 34BCh | | ARRAY | | | | CMP1 widthtype | — |
| 34BCh | 0 | U8 | | RO | no | highest sub-index | — |
| 34BCh | 1 | U8 | | RW | no | Compare1 widthtype 0 | CMP1.WIDTHTYPE 0 |
| 34BCh | 2 | U8 | | RW | no | Compare1 widthtype 1 | CMP1.WIDTHTYPE 1 |
| 34BCh | 3 | U8 | | RW | no | Compare1 widthtype 2 | CMP1.WIDTHTYPE 2 |
| 34BCh | 4 | U8 | | RW | no | Compare1 widthtype 3 | CMP1.WIDTHTYPE 3 |
| 34BCh | 5 | U8 | | RW | no | Compare1 widthtype 4 | CMP1.WIDTHTYPE 4 |
| 34BCh | 6 | U8 | | RW | no | Compare1 widthtype 5 | CMP1.WIDTHTYPE 5 |
| 34BCh | 7 | U8 | | RW | no | Compare1 widthtype 6 | CMP1.WIDTHTYPE 6 |
| 34BCh | 8 | U8 | | RW | no | Compare1 widthtype 7 | CMP1.WIDTHTYPE 7 |
| 34BDh | | ARRAY | | | | CMP1 modes | — |
| 34BDh | 0 | U8 | | RO | no | highest sub-index | — |
| 34BDh | 1 | U8 | | RW | no | Compare1 mode 0 | CMP1.MODE 0 |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|--------------------------------------|
| 34BDh | 2 | U8 | | RW | no | Compare1 mode 1 | CMP1.MODE 1 |
| 34BDh | 3 | U8 | | RW | no | Compare1 mode 2 | CMP1.MODE 2 |
| 34BDh | 4 | U8 | | RW | no | Compare1 mode 3 | CMP1.MODE 3 |
| 34BDh | 5 | U8 | | RW | no | Compare1 mode 4 | CMP1.MODE 4 |
| 34BDh | 6 | U8 | | RW | no | Compare1 mode 5 | CMP1.MODE 5 |
| 34BDh | 7 | U8 | | RW | no | Compare1 mode 6 | CMP1.MODE 6 |
| 34BDh | 8 | U8 | | RW | no | Compare1 mode 7 | CMP1.MODE 7 |
| 34C0h | | ARRAY | | | | Compare0 handling | — |
| 34C0h | 0 | U8 | | RO | no | highest sub-index | — |
| 34C0h | 1 | U16 | | RW | no | Compare0 arm setpoints | CMP0.ARM 0..7 |
| 34C0h | 2 | U16 | | RW | no | Compare0 states | CMP0.STATE 0..7 |
| 34C1h | | ARRAY | | | | Compare1 handling | — |
| 34C1h | 0 | U8 | | RO | no | highest sub-index | — |
| 34C1h | 1 | U16 | | RW | no | Compare1 arm setpoints | CMP1.ARM 0..7 |
| 34C1h | 2 | U16 | | RW | no | Compare1 states | CMP1.STATE 0..7 |
| 3501h | 0 | INT32 | 1:1 | RW | no | Acceleration ramp | DRV.ACC, also see "6083h" (→ p. 996) |
| 3502h | 0 | INT32 | 1:1 | RW | no | Acceleration ramp for homing/jog modes | HOME.ACC |
| 3506h | 0 | INT32 | | | no | Action that hardware enable digital input will perform. | DRV.HWENMODE |
| 3509h | 0 | INT32 | 1000:1 | RO | no | Analog input voltage | AIN.VALUE |
| 3522h | 0 | INT32 | 1:1 | RW | no | Deceleration rate | DRV.DEC, also see "6084h" (→ p. 996) |
| 3524h | 0 | INT32 | 1:1 | RW | no | Deceleration ramp for homing/jog modes | HOME.DEC |
| 352Ah | 0 | INT32 | | RW | no | Direction of movements | DRV.DIR |
| 3533h | 0 | U32 | | RO | no | Resolution of motor encoder | FB1.ENCRES |
| 3534h | 0 | U32 | | RO | no | Mode of EEO connector | DRV.EMUEMODE |
| 3535h | 0 | U32 | | RO | no | Resolution of EEO | DRV.EMUERES |
| 3537h | 0 | U32 | | RO | no | Location of EEO index pulse | DRV.EMUEZOFFSET |
| 353Bh | 0 | INT32 | | RO | no | Selection of the feedback type | FB1.SELECT |
| 3542h | 0 | U32 | 1000:1 | RW | no | Position Control Loop: Proportional Gain | PL.KP |
| 3548h | 0 | U32 | 1000:1 | RW | no | Velocity Control Loop: Proportional Gain | VL.KP |
| 354Bh | 0 | INT32 | 1000:1 | RW | no | Sets the velocity loop velocity feedforward gain value | VL.KVFF |
| 354Dh | 0 | INT32 | 1000:1 | RW | no | Velocity Control Loop: I-Integration Time | VL.KI |
| 3558h | 0 | INT32 | 1000:1 | RO | no | Current Monitor | IL.FB |
| 3559h | 0 | INT32 | 1000:1 | RO | no | Drive Ifold | IL.DIFOLD |
| 355Ah | 0 | INT32 | 1000:1 | RW | no | I2T Warning | IL.FOLDWTHRESH |
| 3562h | 0 | INT32 | | RW | no | Function of Digital Input 1 | DIN1.MODE |
| 3565h | 0 | INT32 | | RW | no | Function of Digital Input 2 | DIN2.MODE |
| 3568h | 0 | INT32 | | RW | no | Function of Digital Input 3 | DIN3.MODE |
| 356Bh | 0 | INT32 | | RW | no | Function of Digital Input 4 | DIN4.MODE |
| 356Eh | 0 | INT32 | 1000:1 | RW | no | Application Peak Current, positive direction | IL.LIMITP |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|-----------------|
| 356Fh | 0 | INT32 | 1000:1 | RW | no | Application Peak Current, negative direction | IL.LIMITN |
| 3586h | 0 | U32 | | RW | no | Sets the motor temperature fault level | MOTOR.TEMPFAULT |
| 3587h | 0 | INT32 | | RW | no | Select Motor Holding Brake | MOTOR.BRAKE |
| 358Eh | 0 | U32 | 1000:1 | RW | no | Motor Continuous Current Rating | MOTOR.ICONT |
| 358Fh | 0 | U32 | 1000:1 | RW | no | Motor Peak Current Rating | MOTOR.IPEAK |
| 3593h | 0 | U32 | 1000:1 | RW | no | Sets the torque constant of the motor | MOTOR.KT |
| 3596h | 0 | U32 | 1000:1 | RO | no | Sets the proportional gain of the d-component current PI-regulator as a percentage of IL.KP | IL.KPDRATIO |
| 3598h | 0 | INT32 | 1000:1 | RW | no | Absolute Gain of Current Control loop | IL.KP |
| 359Ch | 0 | U32 | | RW | no | Sets the motor phase. | MOTOR.PHASE |
| 359Dh | 0 | U32 | | RW | no | Sets the number of motor poles | MOTOR.POLES |
| 35A3h | 0 | U32 | | RW | no | Sets the maximum motor speed | MOTOR.VMAX |
| 35A4h | 0 | INT32 | 1000:1 | RW | no | Maximum motor current | IL.MIFOLD |
| 35ABh | 0 | U32 | 1000:1 | RW | no | Sets the motor inertia | MOTOR.INERTIA |
| 35AFh | 0 | U32 | | RW | no | Sets the digital output 1 mode | DOOUT1.MODE |
| 35B2h | 0 | U32 | | RW | no | Sets the digital output 2 mode | DOOUT2.MODE |
| 35B4h | 0 | INT32 | | RW | no | Operating Mode | DRV.OPMODE |
| 35B8h | 0 | U32 | | RW | no | Table number for motion task | MT.TNUM |
| 35B9h | 0 | INT32 | | RW | no | Control for Motion Task 0 | MT.CNTL |
| 35BCh | 0 | INT32 | | RW | no | Next Task Number for Motion Task 0 | MT.MTNEXT |
| 35BDh | 0 | U32 | | RW | no | Time to next motion task | MT.TNEXT |
| 35C2h | 0 | INT32 | | RW | no | Select regen resistor | REGEN.REXT |
| 35C5h | 0 | INT32 | 1:1 | RO | no | Actual Following Error | PL.ERR |
| 35C6h | 0 | INT32 | 1:1 | RW | no | In-Position Window (profile position mode) | MT.TPOSWND |
| 35C7h | 0 | INT32 | 1:1 | RW | no | Max. Following Error | PL.ERRFTHRESH |
| 35CAh | 0 | INT32 | | RW | no | Position Resolution (Numerator) | UNIT.PIN |
| 35CBh | 0 | INT32 | | RW | no | Position Resolution (Denominator) | UNIT.POUT |
| 35CFh | 0 | INT32 | | RW | no | reserved | PL.MODPEN |
| 35D2h | 0 | U32 | | RO | no | Mechanical Position | FB1.MECHPOS |
| 35E2h | 0 | U32 | 1:1 | RW | no | Sets the current limit during homing procedure to a mechanical stop | HOME.IPEAK |
| 35EBh | 0 | INT32 | | WO | no | Save Data in EEPROM | DRV.NVSAVE |
| 35F0h | 0 | INT32 | | WO | no | Set Reference Point | HOME.SET |
| 35FEh | 0 | INT32 | | WO | no | Stop Motion Task | DRV.STOP |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|------------------|
| 35FFh | 0 | U32 | | RW | no | Selects between disable immediately or stop and then disable | DRV.DISMODE |
| 3610h | 0 | INT32 | | RO | no | Ambient Temperature | DRV.TEMPERATURES |
| 3611h | 0 | INT32 | | RO | no | Heat Sink Temperature | DRV.TEMPERATURES |
| 3612h | 0 | INT32 | | RO | no | Motor Temperature | MOTOR.TEMP |
| 3617h | 0 | U32 | 1:1 | RW | no | Undervoltage mode | VBUS.UVMODE |
| 3618h | 0 | INT32 | 1:1 | RO | no | Actual Velocity | VL.FB |
| 361Ah | 0 | INT32 | | RO | no | DC-bus voltage | VBUS.VALUE |
| 361Dh | 0 | U32 | 1000:1 | RW | no | Voltage level for undervoltage fault | VBUS.UVFTHRESH |
| 3622h | 0 | INT32 | 1:1 | RW | no | Max. Velocity | VL.LIMITP |
| 3623h | 0 | INT32 | 1:1 | RW | no | Max. Negative Velocity | VL.LIMITN |
| 3627h | 0 | INT32 | 1:1 | RW | no | Overspeed | VL.THRESH |
| 3629h | 0 | INT32 | 1000:1 | RW | no | SW1 Velocity Scaling Factor | AIN.VSCALE |
| 3637h | 0 | INT32 | 1:1 | RW | no | reserved | PL.MODP1 |
| 3638h | 0 | INT32 | 1:1 | RW | no | reserved | PL.MODP2 |
| 3656h | 0 | U64 | 1:1 | RW | no | Initial feedback position | FB1.ORIGIN |
| 3659h | 0 | INT32 | | RW | no | Type of acceleration setpoint for the system | UNIT.ACCROTARY |
| 365Bh | 0 | INT32 | | RW | no | Presetting for motion task that is processed later | MT.NUM |
| 365Fh | 0 | INT32 | | RW | no | Systemwide Definition of Velocity/Speed | UNIT.VROTARY |
| 3660h | 0 | INT32 | | RW | no | Set Resolution of the Position | UNIT.PROTARY |
| 366Eh | 0 | INT32 | | RW | no | Disable Delaytime with Holding Brake | MOTOR.TBRAKEAPP |
| 366Fh | 0 | INT32 | | RW | no | Enable Delaytime with Holding Brake | MOTOR.TBRAKERLS |
| 3683h | 0 | U16 | | RW | no | Delay for wake and shake timing | WS.TDELAY1 |
| 3685h | 0 | U16 | | RW | no | Sets delay for wake and shake timing | WS.TDELAY2 |
| 36D0h | 0 | U16 | | RW | no | Sets wake and shake current-vector appliance time | WS.T |
| 36D1h | 0 | U32 | 1:1 | RW | no | Sets the minimum movement required for wake and shake | WS.DISTMIN |
| 36D7h | 0 | U32 | 1000:1 | RW | no | Sets homing auto move flag | HOME.AUTOMOVE |
| 36E2h | 0 | U8 | | RW | no | Sets the number of repetitions for wake and shake | WS.NUMLOOPS |
| 36E5h | 0 | U32 | | RW | no | CAN baud rate selection | FBUS.PARAM01 |
| 36E6h | 0 | U32 | | RW | no | pll synchronization | FBUS.PARAM02 |
| 36E7h | 0 | U32 | | RW | no | - | FBUS.PARAM03 |
| 36E8h | 0 | U32 | | RW | no | SYNC surveillance | FBUS.PARAM04 |
| 36E9h | 0 | U32 | | RW | no | - | FBUS.PARAM05 |
| 36EAh | 0 | U32 | | RW | no | - | FBUS.PARAM06 |
| 36EBh | 0 | U32 | | RW | no | - | FBUS.PARAM07 |
| 36ECh | 0 | U32 | | RW | no | - | FBUS.PARAM08 |
| 36EDh | 0 | U32 | | RW | no | - | FBUS.PARAM09 |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|--------------|
| 36EEh | 0 | U32 | | RW | no | - | FBUS.PARAM10 |
| 36F6h | 0 | INT32 | | RW | no | Function of Digital Input 5 | DIN5.MODE |
| 36F9h | 0 | INT32 | | RW | no | Function of Digital Input 6 | DIN6.MODE |
| 36FCh | 0 | U32 | | RW | no | Function of Digital Input 7 | DIN7.MODE |
| 3856h | 0 | INT32 | 1:1 | RW | no | velocity window for profile position mode | MT.TVELWND |

Objects 5000h to 5999h

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|------------------|
| 5000h | 0 | UINT32 | | RW | no | Analog input low-pass filter cutoff frequency. | AIN.CUTOFF |
| 5001h | 0 | UINT32 | | RW | no | Analog input signal dead-band. | AIN.DEADBAND |
| 5002h | 0 | UINT32 | | RW | no | Analog current scale factor. | AIN.ISCALE |
| 5003h | 0 | UINT32 | | RW | no | Analog input offset. | AIN.OFFSET |
| 5009h | 0 | UINT32 | | RW | no | Analog current scale factor. | AOUT.ISCALE |
| 500Bh | 0 | UINT32 | | RW | no | Analog output offset. | AOUT.OFFSET |
| 5013h | 0 | UINT32 | | RW | no | Controls how often the excitation is updated. | BODE.EXCITEGAP |
| 5015h | 0 | UINT32 | | RW | no | Current command value used during the Bode procedure. | BODE.IAMP |
| 5016h | 0 | UINT32 | | RW | no | Sets whether the excitation uses current or velocity excitation type. | BODE.INJECTPOINT |
| 5019h | 0 | UINT32 | | RW | no | Length of the PRB signal before it repeats. | BODE.PRBDDEPTH |
| 5060h | 0 | UINT32 | | RW | no | Sets the fault relay mode. | DOUT.RELAYMODE |
| 5080h | 0 | UINT32 | | RW | no | Default state of the software enable. | DRV.ENDEFAULT |
| 5083h | 0 | UINT32 | | RW | no | Continuous rated current value. | DRV.ICONT |
| 5084h | 0 | UINT32 | | RW | no | Peak rated current value. | DRV.IPEAK |
| 5085h | 0 | UINT32 | | RW | no | Current that will be used during the DRV.ZERO procedure. | DRV.IZERO |
| 508Ch | 0 | UINT32 | | RW | no | Number of Biss Sensor (Position) Bits for the BiSS Mode C encoder in use. | FB1.BISSBITS |
| 508Fh | 0 | UINT32 | | RW | no | Initial feedback value as signed or unsigned. | FB1.INITSIGNED |
| 5096h | 0 | UINT32 | | RW | no | Current value used during the phase finding procedure (PFB.PFIND=1) | FB1.PFINDCMDU |
| 5097h | 0 | UINT32 | | RW | no | Number of feedback poles. | FB1.POLES |
| 5099h | 0 | UINT32 | | RW | no | Resolver nominal transformation ratio. | FB1.RESKTR |
| 509Ah | 0 | UINT32 | | RW | no | Electrical degrees of phase lag in the resolver. | FB1.RESREFPHASE |
| 509Ch | 0 | UINT32 | | RW | no | Controls tracking calibration algorithm. | FB1.TRACKINGCAL |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|----------------|
| 50B1h | 0 | UINT32 | | RW | no | Number of successful synchronized cycles needed to lock the PLL. | FBUS.PLLTHRESH |
| 50BBh | 0 | UINT32 | | RW | no | Denominator of the electronic gearing ratio; active in opmode 2 (position) only. | GEAR.IN |
| 50BCh | 0 | UINT32 | | RW | no | Electronic gearing mode; active in opmode 2 (position) only. | GEAR.MODE |
| 50BEh | 0 | UINT32 | | RW | no | Numerator of the electronic gearing ratio; active in opmode 2 (position) only. | GEAR.OUT |
| 50C5h | 0 | UINT32 | | RW | no | Homing direction | HOME.DIR |
| 50CBh | 0 | UINT32 | | RW | no | Homing mode | HOME.MODE |
| 50E2h | 0 | UINT32 | | RW | no | Current loops fieldbus injected feed-forward gain | IL.KBUSFF |
| 50FBh | 0 | UINT32 | | RW | no | Motor pitch. | MOTOR.PITCH |
| 50FEh | 0 | UINT32 | | RW | no | Type of thermal resistor inside the motor. | MOTOR.RTYPE |
| 5104h | 0 | UINT32 | | RW | no | Motor type. | MOTOR.TYPE |
| 510Eh | 0 | UINT32 | | RW | no | Motion task to be triggered after an emergency stop procedure; active in opmode 2 (position) only. | MT.EMERGMT |
| 5121h | 0 | UINT32 | | RW | no | Type of following error warning and fault usage. | PL.ERRMODE |
| 5128h | 0 | UINT32 | | RW | no | Feedback source for the position loop. | PL.FBSOURCE |
| 5175h | 0 | UINT32 | | RW | no | Service motion current 1; active in opmode 0 (torque) only. | SM.I1 |
| 5176h | 0 | UINT32 | | RW | no | Service motion current 2; active in opmode 0 (torque) only. | SM.I2 |
| 5177h | 0 | UINT32 | | RW | no | Service motion mode. | SM.MODE |
| 5179h | 0 | UINT32 | | RW | no | Service motion time 1. | SM.T1 |
| 517Ah | 0 | UINT32 | | RW | no | Service motion time 2. | SM.T2 |
| 517Eh | 0 | UINT32 | | RW | no | Enables and disables software travel limit switches. | SWLS.EN |
| 5184h | 0 | UINT32 | | RW | no | Linear acceleration/deceleration units. | UNIT.ACCLINEAR |
| 5187h | 0 | UINT32 | | RW | no | Linear position units. | UNIT.PLINEAR |
| 518Ah | 0 | UINT32 | | RW | no | Linear velocity units. | UNIT.VLINEAR |
| 518Eh | 0 | UINT32 | | RW | no | Voltage level for over voltage warning. | VBUS.OVWTHRESH |
| 51AEh | 0 | UINT32 | | RW | no | Feedback source for the velocity loop; active in opmodes 1 (velocity) and 2 (position) only. | VL.FBSOURCE |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|---------------------|
| 51B0h | 0 | UINT32 | | RW | no | Mode of velocity generation (Observer, d/dt); active in opmodes 1 (velocity) and 2 (position) only. | VL.GENMODE |
| 51B3h | 0 | UINT32 | | RW | no | Scales the observer velocity signal; active in opmodes 1 (velocity) and 2 (position) only. | VL.KO |
| 51B8h | 0 | UINT32 | | RW | no | Ratio of the estimated load moment of inertia relative to the motor moment of inertia; active in opmodes 1 and 2 only. | VL.LMJR |
| 51BAh | 0 | UINT32 | | RW | no | Bandwidth of the observer in Hz. | VL.OBSBW |
| 51BBh | 0 | UINT32 | | RW | no | Observer operating mode. | VL.OBSMODE |
| 51CBh | 0 | UINT32 | | RW | no | Filter mode for Digital In 1. | DIN1.FILTER |
| 51CCh | 0 | UINT32 | | RW | no | Filter mode for Digital In 2. | DIN2.FILTER |
| 51CDh | 0 | UINT32 | | RW | no | Filter mode for Digital In 3. | DIN3.FILTER |
| 51CEh | 0 | UINT32 | | RW | no | Filter mode for Digital In 4. | DIN4.FILTER |
| 51CFh | 0 | UINT32 | | RW | no | Filter mode for Digital In 5. | DIN5.FILTER |
| 51D0h | 0 | UINT32 | | RW | no | Filter mode for Digital In 6. | DIN6.FILTER |
| 51D1h | 0 | UINT32 | | RW | no | Filter mode for Digital In 7. | DIN7.FILTER |
| 51E7h | 0 | UINT32 | | RW | no | Modbus User Units Input parameter | MODBUS.PIN |
| 51E8h | 0 | UINT32 | | RW | no | Modbus User Units Output parameter. | MODBUS.POUT |
| 51E9h | 0 | UINT32 | | RW | no | Feedback Resolution (per rev) over Modbus. | MODBUS.PSCALE |
| 51ECh | 0 | UINT32 | | RW | no | Secondary feedback (FB2) resolution. | FB2.ENCRECRES |
| 51EDh | 0 | UINT32 | | RW | no | Mode for the second feedback inputs and high speed digital inputs. | FB2.MODE |
| 51EEh | 0 | UINT32 | | RW | no | Source for the second feedback input. | FB2.SOURCE |
| 51EFh | 0 | UINT32 | | RW | no | Brake apply timeout for vertical axis. | MOTOR.TBRAKETO |
| 51F0h | 0 | UINT32 | | RW | no | i.p. | MODBUS.MSGLOG |
| 520Ch | 0 | UINT32 | | RW | no | Scaling mode for Modbus values. | MODBUS.SCALING |
| 520Dh | 0 | UINT32 | | RW | no | Encoder output pulse width for modes 6 to 7. | DRV.EMUEPULSE-WIDTH |
| 520Eh | 0 | UINT32 | | RW | no | Enable/disable motor velocity vs. maximum emulated encoder velocity monitoring function. | DRV.EMUECHECK-SPEED |
| 5251h | 0 | UINT32 | | RW | no | Analog input deadband mode. | AIN.DEADBANDMODE |
| 5252h | 0 | UINT32 | | RW | no | Analog input mode | AIN.MODE |
| 5253h | 0 | UINT32 | | RW | no | Direction of IOs from X9. | DIO10.DIR |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|------------------|
| 5254h | 0 | UINT32 | | RW | no | Inverting the output voltage of the IO, when in the output direction. | DIO10.INV |
| 5255h | 0 | UINT32 | | RW | no | Direction of IOs from X9. | DIO11.DIR |
| 5256h | 0 | UINT32 | | RW | no | Inverting the output voltage of the IO, when in the output direction. | DIO11.INV |
| 5257h | 0 | UINT32 | | RW | no | Direction of IOs from X9. | DIO9.DIR |
| 5258h | 0 | UINT32 | | RW | no | Inverting the output voltage of the IO, when in the output direction. | DIO9.INV |
| 5259h | 0 | UINT32 | | RW | no | Fault Action for Fault 130. | FAULT130.ACTION |
| 525Ah | 0 | UINT32 | | RW | no | Fault Action for Fault 131. | FAULT131.ACTION |
| 525Bh | 0 | UINT32 | | RW | no | Fault Action for Fault 132. | FAULT132.ACTION |
| 525Ch | 0 | UINT32 | | RW | no | Fault Action for Fault 133. | FAULT134.ACTION |
| 525Dh | 0 | UINT32 | | RW | no | Fault Action for Fault 702. | FAULT702.ACTION |
| 525Eh | 0 | UINT32 | | RW | no | Method of acquiring IP Address. | IP.MODE |
| 525Fh | 0 | UINT32 | | RW | no | Load inertia. | LOAD.INERTIA |
| 5260h | 0 | UINT32 | | RW | no | Motor back EMF constant. | MOTOR.KE |
| 5261h | 0 | UINT32 | | RW | no | Changing voltage thresholds. | VBUS.HALFVOLT |
| 5262h | 0 | UINT32 | | RW | no | Direction for the second feedback input (X9 and X7). | FB2.DIR |
| 5263h | 0 | UINT32 | | RW | no | Feedback for handwheel operation. | DRV.HANDWHEELSRC |
| 5264h | 0 | UINT32 | | RW | no | Delay time between inactive Hardware Enable input and drive disable. | DRV.HWENDELAY |
| 5265h | 0 | UINT32 | | RW | no | Index into the Current Loop Gain Scheduling Table. | IL.KPLOOKUPINDEX |
| 5266h | 0 | UINT32 | | RW | no | Value of the current loop gain scheduling index. | IL.KPLOOKUPVALUE |
| 5267h | 0 | UINT32 | | RW | no | Fault Action for Fault 451. | FAULT451.ACTION |
| 5268h | 0 | UINT32 | | RW | no | Brake Immediately in the case of a drive disable. | MOTOR.BRAKEIMM |
| 5352h | 0 | UINT16 | | RW | no | Amount of time a communication error must be present before an W&S-fault is thrown. | WS.CHECKT |
| 535Ch | 0 | UINT16 | | RW | no | Sets the calming time of the motor for Wake & Shake mode 1. | WS.TSTANDSTILL |
| 535Dh | 0 | UINT16 | | RW | no | Time for the ramp up current in Wake & Shake mode 1. | WS.TIRAMP |
| 5360h | 0 | UINT16 | | RW | no | Rotor time constant. | MOTOR.IMTR |
| 5361h | 0 | UINT8 | | RW | no | Sets the feedback source for the current loop for MOTOR.TYPE4. | IL.FBSOURCE |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|----------------------|
| 5362h | 0 | UINT32 | | RW | no | The direct-axis current set point used for induction machine closed-loop control. | MOTOR.IMID |
| 538Bh | 0 | UINT16 | | RW | no | | DRV.EMUESTEPMODE () |
| 538Ch | 0 | UINT16 | | RW | no | | DRV.EMUESTEPSTATUS |
| 538Dh | 0 | UINT16 | | RW | no | | DRV.EMUESTEPVMAX |
| 538Fh | 0 | INT8 | | RW | no | Compare engine 0 source | CMP0.SOURCE |
| 5390h | 0 | INT8 | | RW | no | Compare engine 1 source | CMP1.SOURCE |
| 5394h | 0 | U16 | | RW | no | Compare engine 0 output mask | CMP0.OUTMASK |
| 539Bh | 0 | U16 | | RW | no | Compare engine 1 output mask | CMP1.OUTMASK |
| 53A6h | 0 | U8 | | RW | no | Compare engine 0 modulo enable | CMP0.MODEN |
| 53ADh | 0 | U8 | | RW | no | Compare engine 1 modulo enable | CMP1.MODEN |
| 53B1h | 0 | U32 | | RW | no | Compare engine 0 advance | CMP0.ADVANCE |
| 53B2h | 0 | U32 | | RW | no | Compare engine 1 advance | CMP1.ADVANCE |
| 53C7h | 0 | UINT32 | | RW | no | Sets the fault display mode | DRV.FAULTDISPLAYMODE |
| 53D5h | 0 | UINT32 | | RW | no | Sets the delay time for PL.CMD | PL.PDELAY |
| 53D6h | 0 | UINT32 | | RW | no | Sets the delay time for the velocity feedforward integrator component | VL.FFDELAY |
| 53D7h | 0 | INT8 | | RW | no | Allows a surface permanent magnet motor to operate as an interior permanent magnet motor | MOTOR.FIELDWEAKENING |
| 5403h | 0 | UINT32 | | RW | no | Toggles between HOME.IPEAK and current loop limits during homing | HOME.IPEAKACTIVE |
| 5404h | 0 | UINT32 | | RW | no | Scaling factor (numerator) for the command DRV.EMUESTEPCMD | DRV.EMUESTEPCMDPIN |
| 5405h | 0 | UINT32 | | RW | no | Scaling factor (denominator) for the command DRV.EMUESTEPCMD | DRV.EMUESTEPCMDPOUT |
| 5406h | 0 | UINT32 | | RW | no | Sets the target position window for the homing procedure; active in opmode 2 (position) only | HOME.TPOSWND |

21.2.5 Profile specific SDOs

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|------------------------|--------------|
| 6040h | 0 | U16 | | WO | yes | Control word | — |
| 6041h | 0 | U16 | | RO | yes | Status word | — |
| 605Ah | 0 | INT16 | | RW | no | Quick stop option code | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|------------------|
| 6060h | 0 | INT8 | | RW | yes | Modes of Operation | — |
| 6061h | 0 | INT8 | | RO | yes | Modes of Operation Display | — |
| 6063h | 0 | INT32 | | RO | yes | Position actual value (increments) | — |
| 6064h | 0 | INT32 | 1:1 | RO | yes | Position actual value (position units) | PL.FB |
| 6065h | 0 | U32 | 1:1 | RW | no | Following error window | PL.ERRFTHRESH |
| 606Bh | 0 | INT32 | 1:1 | RO | no | Velocity demand value | VL.CMD |
| 606Ch | 0 | INT32 | 1000:1 | RO | yes | Velocity actual value (PDO in RPM) | VL.FB |
| 606Dh | 0 | U16 | | RW | yes | Velocity window | |
| 606Eh | 0 | U16 | | RW | yes | Velocity window time | |
| 6071h | 0 | INT16 | | RW | yes* | Target torque | — |
| 6072h | 0 | U16 | | RW | yes* | Max torque | — |
| 6073h | 0 | U16 | | RW | no | Max current | |
| 6077h | 0 | INT16 | | RO | yes | Torque actual value | DRV.ICONT |
| 607Ah | 0 | INT32 | 1:1 | RW | yes | Target position | MT.P |
| 607Ch | 0 | INT32 | 1:1 | RW | no | Reference offset | HOME.P |
| 607Dh | | ARRAY | | | | Software position limit | |
| 607Dh | 0 | U8 | | RO | no | highest sub-index | |
| 607Dh | 1 | INT32 | 1:1 | RW | no | Software position limit 1 | SWLS.LIMIT0 |
| 607Dh | 2 | INT32 | 1:1 | RW | no | Software position limit 2 | SWLS.LIMIT1 |
| 6081h | 0 | U32 | 1:1 | RW | yes | Profile Velocity | MT.V |
| 6083h | 0 | U32 | 1:1 | RW | yes | Profile Acceleration | MT.ACC , DRV.ACC |
| 6084h | 0 | U32 | 1:1 | RW | yes | Profile Deceleration | MT.DEC , DRV.DEC |
| 6087h | 0 | U32 | | RW | yes | Torque slope | — |
| 608Fh | | ARRAY | | | | Position encoder resolution | — |
| 608Fh | 0 | U8 | | RO | no | highest sub-index | — |
| 608Fh | 1 | U32 | | RW | no | Encoder increments | — |
| 608Fh | 2 | U32 | | RW | no | Motor revolutions | |
| 6091h | | ARRAY | | | | Gear ratio | — |
| 6091h | 0 | U8 | | RO | no | highest sub-index | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|----------------------|
| 6091h | 1 | U32 | | RW | no | Motor revolution | |
| 6091h | 2 | U32 | | RW | no | Shaft revolutions | |
| 6092h | | ARRAY | | | | Feed constant | — |
| 6092h | 0 | U8 | | RO | no | highest sub-index | — |
| 6092h | 1 | U32 | | RW | no | Feed | UNIT.PIN |
| 6092h | 2 | U32 | | RW | no | Shaft revolutions | — |
| 6098h | 0 | INT8 | | RW | no | Homing type | HOME.MODE , HOME.DIR |
| 6099h | | ARRAY | | | | Homing velocity | — |
| 6099h | 0 | U8 | | RO | no | highest sub-index | — |
| 6099h | 1 | U32 | 1:1 | RW | no | Speed while searching for limit switch | HOME.V |
| 6099h | 2 | U32 | | RW | no | Speed while searching for zero mark | HOME.FEEDRATE |
| 609Ah | 0 | U32 | 1:1 | RW | no | Homing acceleration | HOME.ACC , HOME.DEC |
| 60B1h | 0 | INT32 | 1:1 | RW | yes* | Velocity offset | VL.BUSFF |
| 60B2h | 0 | INT16 | | RW | yes* | Torque offset (PDO only) | |
| 60B8h | 0 | U16 | | RW | yes | Touch probe function | — |
| 60B9h | 0 | U16 | | RW | yes | Touch probe status | — |
| 60BAh | 0 | INT32 | | RW | yes | Touch probe 1 positive edge | — |
| 60BBh | 0 | INT32 | | RW | yes | Touch probe 1 negative edge | — |
| 60BCh | 0 | INT32 | | RW | yes | Touch probe 2 positive edge | — |
| 60BDh | 0 | INT32 | | RW | yes | Touch probe 2 negative edge | — |
| 60C0h | 0 | INT16 | | RW | no | Interpolation sub-mode select | — |
| 60C1h | | ARRAY | | | | Interpolation data record | — |
| 60C1h | 0 | U8 | | RO | no | highest sub-index | — |
| 60C1h | 1 | INT32 | | RW | yes* | Interpolation target position | — |
| 60C1h | 2 | U32 | | RW | yes | Interpolation time | — |
| 60C1h | 3 | INT32 | | RW | yes | Interpolation target velocity | — |
| 60C2h | | RECORD | | | | Interpolation time period | — |
| 60C2h | 0 | U8 | | RO | no | highest sub-index | FBUS.SAMPLEPERIOD |
| 60C2h | 1 | U8 | | RW | no | Interpolation time units | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|---|---------------------------|
| 60C2h | 2 | INT8 | | RW | no | Interpolation time index | — |
| 60C4h | | RECORD | | | | Interpolation data configuration | — |
| 60C4h | 0 | U8 | | RO | no | highest sub-index | — |
| 60C4h | 1 | U32 | | RO | no | Maximum buffer size | — |
| 60C4h | 2 | U32 | | RO | yes | Actual buffer size | — |
| 60C4h | 3 | U8 | | RW | no | Buffer organization | — |
| 60C4h | 4 | U16 | | RW | no | Buffer position | — |
| 60C4h | 5 | U8 | | WO | no | Size of data record | — |
| 60C4h | 6 | U8 | | WO | no | Buffer clear | — |
| 60D0h | | ARRAY | | | | Touch probe source | — |
| 60D0h | 0 | U8 | | RO | no | highest sub-index | - |
| 60D0h | 1 | INT16 | | RW | no | Touch probe 1 source | — |
| 60D0h | 2 | INT16 | | RW | no | Touch probe 2 source | — |
| 60E0h | 0 | UINT16 | | RO | yes* | Positive torque limit value | IL.LIMITP |
| 60E1h | 0 | UINT16 | | RO | yes* | Negative torque limit value | IL.LIMITN |
| 60E4h | | ARRAY | | | | Additional position actual value | — |
| 60E4h | 0 | U8 | | RO | no | highest sub-index | — |
| 60E4h | 1 | INT32 | | RW | no | 1st additional position actual value | — |
| 60E4h | 2 | INT32 | | RW | no | reserved | — |
| 60E4h | 3 | INT32 | | RW | no | 3rd additional position actual value | — |
| 60E8h | | ARRAY | | | | Additional gear ratio - motor shaft revolutions | — |
| 60E8h | 0 | U8 | | RO | no | highest sub-index | — |
| 60E8h | 1 | U32 | | RW | no | 1st additional gear ratio - motor shaft revolutions | — |
| 60E8h | 2 | U32 | | RW | no | reserved | — |
| 60E8h | 3 | U32 | | RW | no | 3rd additional gear ratio - motor shaft revolutions | DS402.3ADDPOSGEARMOTORREV |
| 60E9h | | ARRAY | | | | Additional feed constant - feed | — |

| Index | Sub-index | Data Type | Float Scale | Access | PDO map. | Description | ASCII object |
|-------|-----------|-----------|-------------|--------|----------|--|--|
| 60E9h | 0 | U8 | | RO | no | highest sub-index | — |
| 60E9h | 1 | U32 | | RW | no | 1st additional feed constant - feed | DS402.1ADDPOSFCFEED |
| 60E9h | 2 | U32 | | RW | no | reserved | — |
| 60E9h | 3 | U32 | | RW | no | 3rd additional feed constant - feed | DS402.3ADDPOSFCFEED |
| 60EDh | | ARRAY | | | | Additional gear ratio - driving shaft revolutions | — |
| 60EDh | 0 | U8 | | RO | no | highest sub-index | — |
| 60EDh | 1 | U32 | | RW | no | 1st additional gear ratio - driving shaft revolutions | DS402.1ADDPOSGEARSHAFTREV |
| 60EDh | 2 | U32 | | RW | no | reserved | — |
| 60EDh | 3 | U32 | | RW | no | 3rd additional gear ratio - driving shaft revolutions | DS402.3ADDPOSGEARSHAFTREV |
| 60EEh | | ARRAY | | | | Additional feed constant - driving shaft revolutions | — |
| 60EEh | 0 | U8 | | RO | no | highest sub-index | — |
| 60EEh | 1 | U32 | | RW | no | 1st additional feed constant - driving shaft revolutions | DS402.1ADDPOSFCFSHAFTREV |
| 60EEh | 2 | U32 | | RW | no | reserved | — |
| 60EEh | 3 | U32 | | RW | no | 3rd additional feed constant - driving shaft revolutions | DS402.3ADDPOSFCFSHAFTREV |
| 60F4h | 0 | INT32 | | RO | yes | Following error actual value | PL.ERR |
| 60FCh | 0 | INT32 | | RO | yes | Position demand internal value | PL.CMD |
| 60FDh | 0 | U32 | | RO | yes | Digital inputs | DIN1.MODE TO DIN6.MODE DIN1.MODE TO DIN3.MODE |
| 60FEh | | ARRAY | | | | Digital outputs | |
| 60FEh | 0 | U8 | | RO | no | highest sub-index | — |
| 60FEh | 1 | U32 | | RW | yes | Physical outputs | |
| 60FEh | 2 | U32 | | RW | no | Bit mask | |
| 60FFh | 0 | INT32 | 1000:1 | RW | yes* | Target velocity | VL.CMDU |
| 6502h | 0 | U32 | | RO | no | Supported drive modes | — |

21.3 Object descriptions

The objects in this section are sorted by object number.

21.3.1 Object 1000h: Device Type (DS301)

This object describes the device type (servo drive) and device functionality (DS402 drive profile). Definition:

| | | | | | |
|------------------------|----|------|-----------------------|-----------|---|
| MSB | | | | LSB | |
| Additional information | | | Device profile number | | |
| Mode bits | | Type | | 402d=192h | |
| 31 | 24 | 23 | 16 | 15 | 0 |

The device profile number is DS402, the type is 2 for drives, the mode bits 28 to 31 are manufacturer specific and may be changed from its actual value of 0. A read access delivers 0x00020192 at the moment.

| | |
|---------------|--------------|
| Index | 1000h |
| Name | device type |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | no |

21.3.2 Object 1001h: Error register (DS301)

This object is an error register for the device. The device can map internal errors into this byte. It is a part of an Emergency object.

| | |
|---------------|----------------|
| Index | 1001h |
| Name | Error register |
| Object code | VAR |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | no |

Error reasons to be signaled: If a bit is set to 1 the specified error has occurred. The generic error is signaled at any error situation.

| Bit | Description | Bit | Description |
|-----|---------------|-----|--|
| 0 | generic error | 4 | communication error (overrun, error state) |
| 1 | current | 5 | device profile specific |
| 2 | voltage | 6 | reserved (always 0) |
| 3 | temperature | 7 | manufacturer specific |

21.3.3 Object 1002h: Manufacturer Status Register (DS301)

The manufacturer status register contains important drive informations.

| | |
|---------------|------------------------------|
| Index | 1002h |
| Name | Manufacturer Status Register |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | UNSIGNED32 |
| Default value | no |

The following table shows the bit assignment for the status register:

| Bit | Description | Bit | Description |
|-----|--|-----|---------------------------------------|
| 0 | 1 = Movement (positioning, homing) active | 16 | 1 = Homing move active |
| 1 | reference position set | 17 | reserved |
| 2 | 1 = reference switch high (home-position) | 18 | reserved |
| 3 | 1 = In Position | 19 | 1 = Emergency stop active |
| 4 | reserved | 20 | reserved |
| 5 | reserved | 21 | reserved |
| 6 | reserved | 22 | reserved |
| 7 | Active Disabel activated | 23 | 1 = Homing move finished |
| 8 | Warning active | 24 | Power stage deactivating |
| 9 | 1 = target velocity reached (pp- or pv-Mode) | 25 | 1 = digital input 1 set |
| 10 | reserved | 26 | 1 = digital input 2 set |
| 11 | 1 = Homing error | 27 | 1 = digital input 3 set |
| 12 | reserved | 28 | 1 = digital input 4 set |
| 13 | 1 = Safe Torque Off selected | 29 | 1 = digital input hardware enable set |
| 14 | 1 = Power stage enabled | | |
| 15 | 1 = Error state | 31 | Braking, 1 = set points not accepted |

21.3.4 Object 1003h: Predefined Error Field (DS301)

The object 1003h provides an error history with a maximum size of 10 entries.

Subindex 0 contains the number of errors which have occurred since the last reset of the error history, either by startup of the drive or resetting the error history by writing 0 to subindex 0.

A new Emergency-message is written into subindex 1 shifting the old entries one subindex higher. The old content of subindex 8 is lost.

The UNSIGNED32-information written to the subindexes is defined in the field Error Code in the description of the Emergency Messages (→ p. 968).

| | |
|---------------|---------------------------------|
| Index | 1003h |
| Name | pre-defined Error Field |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | 0 to 10 |
| Default value | 0 |
| Subindex | 1 to 10 |
| Description | Standard error field (→ p. 968) |
| Category | optional |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | no |

21.3.5 Object 1005h: COB-ID of the SYNC Message (DS301)

This object defines the COB-Id of the synchronisation object (SYNC).

| | |
|---------------|-----------------------------|
| Index | 1005h |
| Name | COB-ID for the SYNC message |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | conditional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0x80 |

Bit coded information:

| Bit | Value | Meaning |
|---------------|-------|----------------------------------|
| 31 (MSB) | X | — |
| 30 | 0 | Device not generate SYNC message |
| | 1 | Device generates SYNC message |
| 29 | 0 | 11 Bit ID (CAN 2.0A) |
| | 1 | 29 Bit ID (CAN 2.0B) |
| 28 to 11 | X | — |
| | 0 | if Bit 29=0 |
| 10 to 0 (LSB) | X | Bit 0 to 10 of SYNC COB-ID |

The device does not support the generation of SYNC-messages and only the 11-bit IDs. So the bits 11 to 30 are always 0.

21.3.6 Object 1006h: Communication Cycle Period (DS301)

This object can be used to define the period (in μ s) for the transmission of the SYNC telegram.

| | |
|---------------|-----------------------------------|
| Index | 1006h |
| Name | Period of the communication cycle |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | O |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 00h |

21.3.7 Object 1008h: Manufacturer Device Name (DS301)

The device name consists of four ASCII characters in the form Yzzz, whereby Y stands for the mains voltage (L, M, H or U, e.g. H for High Voltage) zzz stands for the power stage current.

| | |
|---------------|--------------------------|
| Index | 1008h |
| Name | Manufacturer Device Name |
| Object code | VAR |
| Data type | Visible String |
| Category | Optional |
| Access | const |
| PDO mapping | not possible |
| Value range | |
| Default value | no |

21.3.8 Object 1009h: Manufacturer Hardware Version

This object will be supported in the future.

| | |
|---------------|-------------------------------|
| Index | 1009h |
| Name | manufacturer hardware version |
| Object code | VAR |
| Data type | Visible String |
| Category | Optional |
| Access | const |
| PDO mapping | not possible |
| Value range | - |
| Default value | no |

21.3.9 Object 100Ah: Manufacturer Software Version (DS301)

The object contains the manufacturer software version (here: the CANopen-part of the drive firmware).

| | |
|---------------|-------------------------------|
| Index | 100Ah |
| Name | Manufacturer Software Version |
| Object code | VAR |
| Data type | Visible String |
| Category | Optional |
| Access | const |
| PDO mapping | not possible |
| Value range | 0.01 to 9.99 |
| Default value | no |

21.3.10 Object 100Ch: Guard Time (DS301)Response monitoring

The arithmetical product of the Objects 100Ch Guard Time and 100Dh Lifetime Factor is the response monitoring time. The Guard Time is given in milliseconds. The response monitoring is activated with the first Nodeguard object. If the value of the object Guard Time is set to zero, then the response monitoring is inactive.

| | |
|---------------|--|
| Index | 100Ch |
| Name | Guard Time |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | conditional; mandatory, if heartbeat not supported |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED16 |
| Default value | 0 |

21.3.11 Object 100Dh: Lifetime Factor (DS301)

The product of Guard Time and Life Time Factor gives the life time for the nodeguarding protocol. If it's 0, the protocol is not used.

| | |
|---------------|--|
| Index | 100Dh |
| Name | Lifetime Factor |
| Object code | VAR |
| Data type | UNSIGNED8 |
| Category | conditional; (mandatory, if heartbeat not supported) |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | 0 |

21.3.12 Object 1010h: Store Parameters (DS301)

This object supports the saving of parameters to a flash EEPROM. Only the subindex 1 for saving of all parameters, which can also be saved in the parameter files via the GUI, is supported.

| | |
|---------------|-------------------------------|
| Index | 1010h |
| Name | store parameters (DRV.NVSAVE) |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Name | highest sub-index supported |
| Object code | VAR |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value range | 1 |
| Default value | 1 |
| Subindex | 1 |
| Name | save all parameters |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |

Data definition:

| Bit | Value | Meaning |
|---------|-------|--|
| 31 to 2 | 0 | reserved (=0) |
| 1 | 0 | Device does not save parameters autonomously |
| | 1 | Device does save parameters autonomously |
| 0 | 0 | Device does not save parameters on command |
| | 1 | Device does save parameters on command |

By read access to subindex 1 the drive provides information about its storage functionality.

This drive provides a constant value of 1 by read access, i.e. all parameters can be saved by writing to Object 1010 sub 1. In general the drive does not save parameters autonomously with the exception of e.g. the special treatment of the homing of multiturn absolute encoders.

Storing of parameters is only done if a special signature ("save") is written to subindex 1. "save" is equivalent to the unsigned32 - number 65766173h.

21.3.13 Object 1011h: Restore Default Parameters DS301

With this object the default values of parameters according to the communication or device profile are restored. The AKD PDMM gives the possibility to restore all default values.

| | |
|---------------|---|
| Index | 1011h |
| Name | restore default parameters |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Name | highest sub-index supported |
| Object code | VAR |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value range | 1 |
| Default value | 1 |
| Subindex | 1 |
| Name | restore all default parameters (DRV.RSTVAR) |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 (device restores parameter) |

Restoring default parameters to the RAM will be done, if a special signature ("load") is written to subindex 1. "load" has to be transmitted as unsigned32 - number 64616F6Ch.

21.3.14 Object 1012h: COB-ID of the Time Stamp (DS301)

This object defines the COB-ID of the time stamp.

| | |
|---------------|---------------------------|
| Index | 1012h |
| Name | COB-ID for the time stamp |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 100h |

Bit coded information:

| Bit | Content | Value | Meaning |
|---------------|----------|-----------|-------------------------------------|
| 31 (MSB) | consume | 0 | Drive does not consume time message |
| | | 1 | Drive does consume time message |
| 30 | produce | 0 | Drive does not produce time message |
| | | 1 | Drive does produce time message |
| 29 | frame | 0 | Value fixed to 0 |
| 28 to 11 | reserved | – | reserved |
| 10 to 0 (LSB) | CAN-ID | 0h - 800h | COB-ID of the time stamp |

21.3.15 Object 1014h: COB-ID for Emergency Message (DS301)

This object defines the COB-ID of the Emergency message.

| | |
|---------------|---|
| Index | 1014h |
| Name | COB-ID emergency message |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | conditional; mandatory, if Emergency is supported |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 80h + Node - ID |

21.3.16 Object 1016h: Consumer Heartbeat Time

The consumer heartbeat time defines the expected heartbeat cycle time (ms) and must be higher than the corresponding producer heartbeat time configured on the device producing this heartbeat. Monitoring starts after the reception of the first heartbeat. If the consumer heartbeat time is 0 ms the corresponding entry is not used.

| | |
|---------------|-----------------------------|
| Index | 1016h |
| Name | consumer heartbeat time |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value range | 1 |
| Default value | 1 |
| Subindex | 1 |
| Description | Consumer heartbeat time |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value range | unsigned 32 |
| Default value | 0 |

Definition of the entry value of Subindex 1

| | MSB | | | | LSB | |
|------------|----------------------|----|-----------|----|----------------|---|
| Value | reserved (value: 00) | | Node-ID | | heartbeat time | |
| Encoded as | - | | UNSIGNED8 | | UNSIGNED16 | |
| Bit | 31 | 24 | 23 | 16 | 15 | 0 |

21.3.17 Object 1017h: Producer Heartbeat Time

The producer heartbeat time defines the cycle time of the heartbeat in ms. If it's 0, it is not used.

| | |
|---------------|---|
| Index | 1017h |
| Name | Producer heartbeat time |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | conditional; mandatory, if guarding is not supported |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED16 |
| Default value | 0 |

21.3.18 Object 1018h: Identity Object (DS301)

The Identity Object contains general device information.

| | |
|---------------|-----------------------------|
| Index | 1018h |
| Name | Identity Object |
| Object code | RECORD |
| Data type | Identity |
| Category | mandatory |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 1 to 4 |
| Default value | 4 |

Subindex 1 is a unique number for a device manufacturer.

| | |
|---------------|--------------------|
| Subindex | 1 |
| Description | Vendor ID |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0x6Ah (Kollmorgen) |

Subindex 2 contains four ASCII - characters, which determine the voltage range and current class of the device. The voltage range is one character L, M or H for low, medium and high voltage. The next three characters are showing the continuous current of the drive.

| | |
|---------------|----------------------------|
| Subindex | 2 |
| Description | Product Code |
| Category | optional |
| Access | R/O |
| PDO mapping | not possible |
| Value range | e.g. M006 for an MV6 drive |
| Default value | no |

Subindex 3 consists of two revision numbers:

- the major revision number in the upper word containing the CAN-version
- the minor revision number is not used in the AKD PDMM. The firmware version can be retrieved as a string via object 0x100A or as numbers via object 0x2018 subindex 1 to 4.

E.g. a value of 0x0014 0000 means CAN-version 0.20.

| | |
|---------------|-----------------|
| Subindex | 3 |
| Description | Revision Number |
| Category | optional |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | no |

Subindex 4 gives the serial number of the drive. This number contains the following information in it:

- bits 0..14: Board serial number (production in week of year)
- bits 15..20: week of production
- bits 21..24: year of production - 2009
- bits 25..31: ASCII-code of MFR-ID

| | |
|---------------|---------------|
| Subindex | 4 |
| Description | Serial Number |
| Category | optional |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | no |

21.3.19 Object 1026h: OS Prompt

The OS prompt is used to build up an ASCII - communication channel to the drive.

| | |
|---------------|-----------------------------|
| Index | 1026h |
| Name | OS Prompt |
| Object code | ARRAY |
| Data type | UNSIGNED8 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | 2 |

Subindex 1 is used to send one character to the drive.

| | |
|---------------|--------------|
| Subindex | 1 |
| Description | StdIn |
| Category | mandatory |
| Access | W |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | — |

Subindex 2 is used to receive one character from the drive.

| | |
|---------------|--------------|
| Subindex | 2 |
| Description | StdOut |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | 0 |

21.3.20 Objects 1400-1403h: 1st - 4th RxPDO communication parameter (DS301)

1400h to 1403h for RxPDO 1 to 4

| | |
|-------------|----------------------------------|
| Index | 1400h 1401h 1402h 1403h |
| Name | receive PDO parameter |
| Object code | RECORD |
| Data type | PDO CommPar |
| Category | mandatory |

Defined sub-indices

| | | |
|---------------|--|--|
| Subindex | 0 | |
| Name | highest sub-index supported | |
| Data type | UNSIGNED8 | |
| Category | mandatory | |
| Access | R/O | |
| PDO Mapping | not possible | |
| Value Range | 2 | |
| Default Value | 2 | |
| Subindex | 1 | |
| Name | COB-ID used by PDO | |
| Category | mandatory | |
| Access | R/W | |
| PDO Mapping | not possible | |
| Value Range | UNSIGNED32 | |
| Default Value | Index 1400h: 200h + Node-ID Index 1402h: 400h + Node-ID | Index 1401h: 300h + Node-ID Index 1403h: 500h + Node-ID |

Subindex 1 contains the COB-Id of the PDO as a bit coded information:

| Bit | Value | Meaning |
|----------|-------|--|
| 31 | 0 | PDO exists/is valid |
| | 1 | PDO does not exist/is not valid |
| 30 | 0 | RTR allowed on this PDO, not to be used (Can in Automation organisation) |
| | 1 | RTR not allowed on this PDO |
| 29 | 0 | 11 bit-ID (CAN 2.0A) |
| | 1 | 29 bit-ID (CAN 2.0B), not supported |
| 28 to 11 | X | Identifier-bits with 29 bit-ID, not relevant |
| 10 to 0 | X | Bits 10-0 of COB-ID |

| | |
|---------------|-------------------|
| Subindex | 2 |
| Name | transmission type |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED8 |
| Default Value | FFh |

Subindex 2 contains the transmission type of the PDO. There are two ways of setting:

- the value FFh or 255 for event-triggered PDO, which is directly interpreted by reception and taken into actions,
- values from 0 to 240, which cause a SYNC-telegram-controlled interpretation of the PDO contents. Values of 1 to 240 mean, that 0 to 239 SYNC-telegrams are ignored, before one is interpreted. The value 0 means, that only the next SYNC-telegram is interpreted.

21.3.21 Objects 1600-1603h: 1st - 4th RxPDO mapping parameter (DS301)

1600h to 1603h for RxPDO 1 to 4.

| | |
|---------------|---|
| Index | 1600h 1601h 1602h 1603h |
| Name | receive PDO mapping |
| Object Code | RECORD |
| Data Type | PDO Mapping |
| Category | mandatory |
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | 0: PDO is not active 1 - 8: PDO activated, mappings are taken only byte-wise |
| Default Value | PDO1: 1 PDO2: 2 PDO3: 2 PDO4: 2 |
| Subindex | 1 - 8 |
| Name | PDO - mapping for the n-th application object |
| Category | Conditional, depends on number and size of object be mapped |
| Access | R/W |
| PDO Mapping | not possible |

| | |
|---------------|------------|
| Value Range | UNSIGNED32 |
| Default Value | See below |

21.3.22 Objects 1800-1803h: 1st - 4th TxPDO communication parameter (DS301)

1800h to 1803h for TxPDO 1 to 4.

| | |
|-------------|----------------------------------|
| Index | 1800h 1801h 1802h 1803h |
| Name | transmit PDO parameter |
| Object code | RECORD |
| Data type | PDO CommPar |
| Category | mandatory |

| | |
|---------------|-----------------------------|
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 5 |
| Default Value | 5 |

| | |
|---------------|--|
| Subindex | 1 |
| Name | COB-ID used by PDO |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED32 |
| Default Value | Index 1800h: 180h + Node-ID Index 1801h: 280h + Node-ID Index 1802h: 380h + Node-ID Index 1803h: 480h + Node-ID |

| | |
|---------------|-------------------|
| Subindex | 2 |
| Name | transmission type |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED8 |
| Default Value | FFh |

| | |
|---------------|-----------------------|
| Subindex | 3 |
| Name | inhibit time |
| Category | optional |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED16 (n*1/10ms) |
| Default Value | 0h |

| | |
|---------------|--------------|
| Subindex | 4 |
| Name | reserved |
| Category | optional |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | 0 |
| Default Value | 0 |

| | |
|---------------|-----------------------------|
| Subindex | 5 |
| Name | event timer |
| Category | optional |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED16 (0=not used, ms) |
| Default Value | 0h |

Subindex 1 contains the COB-Id of the PDO as a bit coded information:

| Bit-Number | Value | Meaning |
|------------|-------|--|
| 31 | 0 | PDO exists/is valid |
| | 1 | PDO does not exist/is not valid |
| 30 | 0 | RTR allowed on this PDO, not supported |
| | 1 | RTR not allowed on this PDO, not supported |
| 29 | 0 | 11 bit-ID (CAN 2.0A) |
| | 1 | 29 bit-ID (CAN 2.0B), not supported |
| 28 to 11 | X | Identifier-bits with 29 bit-ID, not relevant |
| 10 to 0 | X | Bits 10-0 of COB-ID |

Subindex 2 contains the transmission type of the PDO. There are two ways of setting:

- A value of FFh or 255d for an event-triggered PDO, which is sent immediately after a change in the mapped application objects. Setting of Subindex 3 or 5 has an influence on the sending of a PDO. With Subindex 3 you can configure, in which minimal time the so configured Transmit-PDOs are sent, if PDO-data contents change (reduction of bus-load). With Subindex 5 (event time) a timer is used, which is reset with every event-triggered sending of this PDO. If there is no change of the PDO-content in this time, the PDO is sent caused by this timer event.
- Values from 0 to 240 cause a SYNC-Telegram controlled sending of the PDO.
- Values from 1 to 240 define how often the SYNC-telegram leads to a sending of a PDO.
- The value 0 means, that only the next SYNC-telegram leads to a sending of the so configured PDOs.

21.3.23 Objects 1A00-1A03h: 1st - 4th TxPDO mapping parameter (DS301)

1A00h to 1A03h for TxPDO 1 to 4.

| | |
|---------------|---|
| Index | 1A00h 1A01h 1A02h 1A03h |
| Name | transmit PDO mapping |
| Object Code | RECORD |
| Data Type | PDO Mapping |
| Category | mandatory |
| Subindex | 0 |
| Name | number of mapped application objects in PDO |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | 0: PDO is not active 1 - 8: PDO activated, mappings are taken only byte-wise |
| Default Value | PDO1: 1 PDO2: 2 PDO3: 2 PDO4: 2 |
| Subindex | 1 - 8 |
| Name | PDO - mapping for the n-th application object |
| Category | Conditional, depends on number and size of object be mapped |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED32 |
| Default Value | See below |

21.3.24 Object 1C12h: RxPDO assign (DS301)

This object is used to define the mapping for receive direction of EtherCAT data. Either one of the fixed RxPDO mappings 1701h to 1724h is chosen or 1 to 4 of the free mappings 1600h to 1603h.

| | |
|-------------|--------------|
| Index | 1C12h |
| Name | RxPDO assign |
| Object code | ARRAY |
| Data type | UNSIGNED16 |
| Category | optional |

Defined sub-indices

| | |
|---------------|--------------------------------------|
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | 0: pdo not active, 1-4 PDO activated |
| Default Value | 1 |

| | |
|---------------|--------------------|
| Subindex | 1 to 4 |
| Name | Subindex 001..004 |
| Data type | UNSIGNED8 |
| Category | optional |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED16 |
| Default Value | 1701h via ESI-file |

21.3.25 Object 1C13h: TxPDO assign (DS301)

This object is used to define the mapping for transmit direction of EtherCAT data. Either one of the fixed TxPDO mappings 1B01h to 1B25h is chosen or 1 to 4 of the free mappings 1A00h to 1A03h.

| | |
|-------------|--------------|
| Index | 1C13h |
| Name | TxPDO assign |
| Object code | ARRAY |
| Data type | UNSIGNED16 |
| Category | optional |

Defined sub-indices

| | |
|---------------|--------------------------------------|
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 0: pdo not active, 1-4 PDO activated |
| Default Value | 1 |

| | |
|---------------|--------------------|
| Subindex | 1 to 4 |
| Name | Subindex 001..004 |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED16 |
| Default Value | 1B01h via ESI-file |

21.3.26 Object 2000h: System Warnings

This object is used to show up to three actual warnings with their AKD PDMM- specific warning number.

| | |
|---------------|-------------------------------|
| Index | 2000h |
| Name | System Warnings |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 to 3 |
| Description | DRV.WARNING1 to DRV.WARNINGS3 |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | 0 to 999 |
| Default value | 0 |

21.3.27 Object 2001h: System Faults

This object is used to show up to ten actual faults with their AKD PDMM- specific fault number.

| | |
|---------------|-----------------------------|
| Index | 2001h |
| Name | System Faults |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0xA |
| Default value | 0xA |
| Subindex | 1 to A |
| Description | DRV.FAULT1 to DRV.FAULT10 |
| Mode | independent |

| | |
|---------------|--------------|
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | 0 to 999 |
| Default value | 0 |

21.3.28 Object 2002h: Manufacturer status bytes

This objects delivers the information of the manufacturer status (object 0x1002 sub 0) as four separate, map-able, bytes.

| | |
|---------------|--|
| Index | 2002h |
| Name | Manufacturer status bytes |
| Object code | ARRAY |
| Data type | UNSIGNED8 |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0x4 |
| Default value | 0x4 |
| Subindex | 1 to 4 |
| Description | Manufacturer status byte 1 to Manufacturer status byte 4 |
| Mode | independent |
| Access | R/O |
| PDO mapping | possible |
| Unit | — |
| Value range | 0 to 0xFF |
| Default value | - |

21.3.29 Object 2011h: DRV.RUNTIME in seconds

This object delivers the runtime of the drive in seconds.

| | |
|---------------|------------------------|
| Index | 2011h |
| Name | DRV.RUNTIME in seconds |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Access | R/O |
| Unit | — |
| Value range | UNSIGNED32 |
| Default value | 0 |

21.3.30 Object 2012h: Fault history: Fault numbers

This object delivers the 20 latest entries of the fault numbers of the fault history table. The latest event can be read via sub-index 1. With new events the list is shifted to higher sub-indices.

| | |
|---------------|---|
| Index | 2012h |
| Name | Fault history: Fault numbers |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0x14h |
| Default value | 0x14h |
| Subindex | 1 to 20 |
| Description | Nth-latest entry in fault number list of fault history table (DRV.FAULHIST) |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | 0 - 999 |
| Default value | 0 |

21.3.31 Object 2013h: Fault history: Time stamps

This object delivers the 20 latest entries of the fault time stamps of the fault history table in seconds related to DRV.RUNTIME. The latest event can be read via sub-index 1. With new events the list is shifted to higher sub-indices.

| | |
|---------------|--|
| Index | 2013h |
| Name | Fault history: Time stamps |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0x14h |
| Default value | 0x14h |
| Subindex | 1 to 20 |
| Description | Nth-latest entry in fault time stamp list of fault history table (DRV.FAULTHIST) |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED32 |
| Default value | — |

21.3.32 Object 2014-2017h: 1st-4th Mask 1 to 4 for Transmit-PDO

In order to reduce the bus loading with event-triggered PDOs, masking can be used to switch off the monitoring for individual bits in the PDO. In this way it can be arranged, for instance, that actual position values are only signaled once per turn.

This Object masks the PDO-channels 1 to 4. If only two bytes have been defined in a PDO, then it masks just two bytes, although 4 bytes of mask information have been transmitted.

An activated bit in the mask means that monitoring is active for the corresponding bit in the PDO.

| | |
|---------------|----------------------------------|
| Index | 2014h 2015h 2016h 2017h |
| Name | tx_mask 1 to 4 |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 1 |
| Description | tx_mask1 to 4_low |
| Mode | independent |
| Access | R/W |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED32 |
| Default value | FFFFFFFFh |
| Subindex | 2 |
| Description | tx_mask1 to 4_high |
| Mode | independent |
| Access | R/W |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED32 |
| Default value | FFFFFFFFh |

21.3.33 Object 2018h: Firmware Version

This object gives all information regarding the firmware version.

Example: Firmware version M_01_00_01_005 would show the numbers 1, 0, 1, 5 in the sub-indices 1 to 4.

| | |
|---------------|------------------|
| Index | 2018h |
| Name | firmware version |
| Object code | ARRAY |
| Data type | UNSIGNED16 |
| Subindex | 1 |
| Description | major version |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED16 |
| Default value | 0 |
| Subindex | 2 |
| Description | minor version |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED16 |
| Default value | 0 |
| Subindex | 3 |
| Description | revision |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED16 |
| Default value | 0 |
| Subindex | 4 |
| Description | branch revision |
| Mode | independent |
| Access | R/O |
| PDO mapping | not possible |
| Unit | — |
| Value range | UNSIGNED16 |
| Default value | 0 |

21.3.34 Object 2026h: ASCII Channel

This object is used to build up an ASCII - communication channel to the drive with 4-byte ASCII-strings.

| | |
|---------------|-----------------------------|
| Index | 2026h |
| Name | ASCII Channel |
| Object code | ARRAY |
| Data type | Visible String |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | 2 |

Subindex 1 is used to send four ASCII-characters to the drive.

| | |
|---------------|----------------|
| Subindex | 1 |
| Description | Command |
| Category | mandatory |
| Access | wo |
| PDO mapping | no |
| Value range | Visible String |
| Default value | — |

Subindex 2 is used to receive four characters from the drive.

| | |
|---------------|----------------|
| Subindex | 2 |
| Description | Response |
| Category | mandatory |
| Access | R/O |
| PDO mapping | no |
| Value range | Visible String |
| Default value | - |

21.3.35 Object 204Ch: PV Scaling Factor

This object shall indicate the configured numerator and denominator of the pv scaling factor. The pv scaling factor serves to modify the resolution or directing range of the specified setpoint. It is also included in calculation of the vl velocity demand, and vl velocity actual value. It does not influence the velocity limit function and the ramp function. The value shall have no physical unit and shall be given in the range from -32 768 to +32 767, but the value of 0 shall not be used.

The velocity scaling factor is only active, when bit 4 of FBUS.PARAM05 is set to 1. Otherwise velocities are scaled as 1/1000 rpm.

| | |
|---------------|-------------------------------|
| Index | 204Ch |
| Name | pv scaling factor |
| Object code | ARRAY |
| Data type | INTEGER32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | no |
| Subindex | 1 |
| Description | pv scaling factor numerator |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | +1 |
| Subindex | 2 |
| Description | pv scaling factor denominator |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | +1 |

21.3.36 Object 2071h: Target Current

This parameter can be used alternatively to the DS402 parameter 6071h and is the input to the torque controller. The value is scaled in mA (milli Amperes).

| | |
|---------------|--------------------------------------|
| Index | 2071h |
| Name | Target current |
| Object code | VAR |
| Data type | INTEGER 32 |
| Category | optional |
| Access | RW |
| PDO mapping | possible |
| Value range | depends on DRV.IPEAK and MOTOR.IPEAK |
| Default value | 0 |

21.3.37 Object 2077h: Current Actual Value

This parameter can be used alternatively to the DS402 parameter 6077h. The value is scaled in mA (milli Amperes).

| | |
|---------------|--------------------------------------|
| Index | 2077h |
| Name | Current actual value |
| Object code | VAR |
| Data type | INTEGER 32 |
| Category | optional |
| Access | RO |
| PDO mapping | possible |
| Value range | depends on DRV.IPEAK and MOTOR.IPEAK |
| Default value | 0 |

21.3.38 Object 20A0h: Latch position 1, positive edge

This object is used to output the position or a time, depending on CAP0.MODE, at which the first positive edge occurred on a signal, which can be configured with CAP0.TRIGGER. The latch enable must be active for that purpose(see object 20A4 and 20A5). With CAP0.MODE = 3 the latched position of the encoder index pulse is transferred via this object.

| | |
|---------------|---|
| Index | 20A0h |
| Name | Latch position 1 positive edge CAP0.PLFB, Time capture CAP0.T |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |

| | |
|---------------|---|
| Default value | 0 |
|---------------|---|

21.3.39 Object 20A1h: Latch position 1, negative edge

This object is used to output the position or a time, depending on CAP0.MODE, at which the first negative edge occurred on a signal, which can be configured with CAP0.TRIGGER. The latch enable must be active for that purpose(see object 20A4 and 20A5).

| | |
|---------------|---|
| Index | 20A1h |
| Name | Latch position 1 negative edge CAP0.PLFB, Time capture CAP0.T |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |
| Default value | 0 |

21.3.40 Object 20A2h: Latch position 2, positive edge

This object is used to output the position or a time, depending on CAP1.MODE, at which the first positive edge occurred on a signal, which can be configured with CAP1.TRIGGER. The latch enable must be active for that purpose(see object 20A4 and 20A5).

| | |
|---------------|---|
| Index | 20A2h |
| Name | Latch position 2 positive edge CAP1.PLFB, Time capture CAP1.T |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |
| Default value | 0 |

21.3.41 Object 20A3h: Latch position 2, negative edge

This object is used to output the position or a time, depending on CAP1.MODE, at which the first negative edge occurred on a signal, which can be configured with CAP1.TRIGGER. The latch enable must be active for that purpose(see object 20A4 and 20A5).

| | |
|---------------|---|
| Index | 20A3h |
| Name | Latch position 2 negative edge CAP1.PLFB, Time capture CAP1.T |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |
| Default value | 0 |

21.3.42 Object 20A4h: Latch Control Register

The latch control register is used to enable the latch monitoring of the capture engines 0 and 1. The latch is enabled with a 1 signal and disabled with a 0 signal. Whether or not a latch event has occurred can be recognised by the latch status register (object 20A5).

| | |
|---------------|------------------------|
| Index | 20A4h |
| Name | Latch Control Register |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | rww |
| PDO mapping | possible |
| Value range | 0 to 15 |
| Default value | 0 |

| Bit | Value (bin) | Value (hex) | Description |
|----------|-------------------|-------------|---------------------------------------|
| 0 | 00000000 00000001 | xx01 | Enable extern latch 1 (positive rise) |
| 1 | 00000000 00000010 | xx02 | Enable extern latch 1 (negative rise) |
| 2 | 00000000 00000100 | xx04 | Enable extern latch 2 (positive rise) |
| 3 | 00000000 00001000 | xx08 | Enable extern latch 2 (negative rise) |
| 4 to 7 | | | Reserve |
| 8 | 00000001 00000000 | 01xx | Read external latch 1 (positive rise) |
| 9 | 00000010 00000000 | 02xx | Read external latch 1 (negative rise) |
| 10 | 00000011 00000000 | 03xx | Read external latch 2 (positive rise) |
| 11 | 00000100 00000000 | 04xx | Read external latch 2 (negative rise) |
| 12 to 15 | | | Reserve |

21.3.43 Object 20A5h: Latch Status Register

The latch status register is used to look for the states of the capture engines 0 and 1.

| | |
|---------------|-----------------------|
| Index | 20A5h |
| Name | Latch Status Register |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | rwr |
| PDO mapping | possible |
| Value range | - |
| Default value | 0 |

| Bit | Value (bin) | Value (hex) | Description |
|--------|----------------------|-------------|--|
| 0 | 00000000 00000001 | zz01 | External latch 1 valid (positive rise) |
| 1 | 00000000 00000010 | zz02 | External latch 1 valid (negative rise) |
| 2 | 00000000 00000100 | zz04 | External latch 2 valid (positive rise) |
| 3 | 00000000 00001000 | zz08 | External latch 2 valid (negative rise) |
| 4 to 7 | | | Reserve |
| 8 | 00000001 00000000 | z1zz | Acknowledge value external latch 1 (positive rise) |
| 9 | 00000010 00000000 | z2zz | Acknowledge value external latch 1 (negative rise) |
| 10 | 00000011 00000000 | z3zz | Acknowledge value external latch 2 (positive rise) |
| 11 | 00000100 00000000 | z4zz | Acknowledge value external latch 2 (negative rise) |
| 12 | 00010000 00000000 | 1zzz | State Digital Input 4 |
| 13 | 00100000 00000000 | 2zzz | State Digital Input 3 |
| 14 | 01000000 00000000 | 4zzz | State Digital Input 2 |
| 15 | 10000000 00000000 | 8zzz | State Digital Input 1 |

21.3.44 Object 20A6h: Latch position 1, positive or negative edge

This object is used to output the position or a time, depending on CAP0.MODE, at which the first positive or negative edge occurred on a signal, that can be configured with CAP0.TRIGGER. Latch enable must be active for that purpose (see object 20A4 and 20A5).

| | |
|---------------|---|
| Index | 20A6h |
| Name | Latch position 1 positive or negative CAP0.PLFB |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | ro |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |
| Default value | 0 |

21.3.45 Object 20A7h: Latch position 2, positive or negative edge

This object is used to output the position or a time, depending on CAP1.MODE, at which the first positive or negative edge occurred on a signal, that can be configured with CAP1.TRIGGER. Latch enable must be active for that purpose (see object 20A4 and 20A5).

| | |
|---------------|---|
| Index | 20A7h |
| Name | Latch position 2 positive or negative CAP1.PLFB |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | ro |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Float scaling | var |
| Default value | 0 |

21.3.46 Object 20B8h: Reset of changed input information

This object is used in PDOs to reset the state change information for the digital inputs shown in the Bits 24 to 30 in the object 60FD. Bit 0 to 6 are used to reset the information of the digital input 1 to 7.

| | |
|---------------|------------------------------------|
| Index | 20B8h |
| Name | Reset of changed input information |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | rw |
| PDO mapping | possible |
| Value range | UNSIGNED16 |
| Default value | 0 |

21.3.47 Object 345Ah: Brake Control

These objects implement the possibility to control the brake directly, overriding the drive logic. When the brake state is controlled by the fieldbus, the drive state (enabled, disabled, faulted) will have no effect on the brake - the fieldbus will be in control.

NOTICE

Applying or releasing the brake at the wrong time can be a safety hazard and can destroy your mechanic as well as drive or motor. Unexpected behaviour might be possible. It is the responsibility of the customer using this mode to use this function appropriately.

When fieldbus control is disabled, the drive will control the brake as defined by existing AKD PDMM brake related parameters. As soon as fieldbus control is enabled, the Brake Command received over the field bus will take effect. So, if the Brake Command is set to APPLY and the current state is RELEASE, the brake will begin to apply .

The default value of the fieldbus control will be disabled, so that the drive is always in control until the fieldbus is operational. It is recommended that this bit remain 0 except for special operating conditions where the fieldbus will control the brake. When fieldbus communication is lost, the drive will regain control of the brake if the fieldbus had previously taken control.

| Enable Fieldbus Control | Serious Failure condition present | Brake Command | Fieldbus Control Status | Controlled by... | Final Brake State |
|-------------------------|-----------------------------------|---------------|-------------------------|------------------|-------------------|
| 0 | x | x | 0 | Drive | Drive |
| 1* | no | 0 | 1 | Fieldbus | Applied |
| 1* | no | 1 | 1 | Fieldbus | Released |
| x | yes | any | 0 | Drive | Drive |

1* indicates that a rising edge was seen since the last time the drive applied the brake

| | |
|-------------|---------------|
| Index | 345Ah |
| Name | Brake Control |
| Object code | ARRAY |
| Data type | UNSIGNED16 |
| Category | optional |

Defined sub-indices

| | |
|---------------|-----------------------------|
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 2 |
| Default Value | 2 |

| | |
|---------------|-----------------------|
| Subindex | 1 |
| Name | Brake Control Command |
| Category | optional |
| Access | R/W |
| PDO Mapping | possible |
| Value Range | UNSIGNED16 |
| Default Value | 0 |

With subindex 1 the brake is controlled. Bit definition:

| Bit | Name | Description |
|-----|-------------------------|--|
| 0 | Enable fieldbus control | 0 - brake is not controlled via this object 1 - enable fieldbus control via this object. This function works edge triggered, i.e. this bit has to have a 0 -> 1 transition to activate the brake control functionality. After a fault the functionality is reset and has to be activated again. The activation can be controlled by subindex 2 bit 0. |
| 1 | Brake Command | This command bit is only active, if the functionality was activated via bit 0. The function is as follows: 0 - apply the brake 1 - release the brake |

| | |
|---------------|-----------------------|
| Subindex | 2 |
| Name | Brake Status Response |
| Category | optional |
| Access | R/O |
| PDO Mapping | possible |
| Value Range | UNSIGNED16 |
| Default Value | 0 |

With subindex 2 the brake status can be checked. Bit definition:

| Bit | Name | Description |
|-----|-------------------------|---|
| 0 | Fieldbus control Status | 0 - brake control via 0x345A is disabled or not possible due to drive failure. 1 - enable fieldbus control via this object. This function works edge triggered, i.e. this bit has to have a 0 -> 1 transition to activate the brake control functionality. After a fault the functionality is reset and has to be activated again. The activation can be controlled by subindex 2 bit 0. |
| 1 | Brake Status | 0 - apply the brake 1 - release the brake Note: When the brake is applied or released, there is a time delay MOTOR.TBRAKEAPP or MOTOR.TBRAKEREL, after the receipt of the command before this status bit changes. The status is always reported: it is not affected by fieldbus control. |

| | | |
|---|------------------|---|
| 2 | STO Status | 0 - STO is not active (drive may be enabled) 1 - STO is active (drive can not be enabled) |
| 3 | HW Enable Status | 0 - HW enable is disabled, drive function can not be enabled 1 - HW enable is enabled, drive function can be enabled |

21.3.48 Object 3474h: Parameters for digital inputs

This set of objects is used to set extended parameters for some digital input functions. The parameters can be used for different DINx.MODEs. Therefore the scaling might be different or no scaling is used at all.

Two subindices build an access object to one of these parameters, because they are 64-bit numbers internally, e.g. object 3474 sub 1 gives access to the low 32 bits of DIN1.PARAM whereas 3474 sub 8 sub 4 gives access to the high 32 bits.

If access to the whole 64 bit number is needed the higher bits must be written first. The access to the lower 32 bits then writes the parameter. If the to be written value fits into 32 bit, only the lower part needs to be written. The most-significant bit is then taken as sign-bit for the number.

| | |
|---------------|--|
| Index | 3474h |
| Name | DINx.PARAM |
| Object code | Array |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0xE |
| Default value | 0xE |
| Subindex | 1 to 7 1 to 3 |
| Description | DINx.PARAM low 32 bits, x = 1 .. 7 1 .. 3 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0 |
| Subindex | 8 to 0xE 4 to 6 |
| Description | DINx.PARAM high 32 bits, x = 1 .. 7 1 .. 3 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0 |

21.3.49 Object 3475h: Parameters for digital outputs

This set of objects is used to set extended parameters for some digital output functions. The parameters can be used for different DOUTx.MODEs. Therefore the scaling might be different or no scaling is used at all.

Two subindices build an access object to one of these parameters, because they are 64-bit numbers internally, e.g. object 3475 sub 1 gives access to the low 32 bits of DOUT1.PARAM whereas 3475 sub 32 gives access to the high 32 bits.

If access to the whole 64 bit number is needed the higher bits must be written first. The access to the lower 32 bits then writes the parameter. If the to be written value fits into 32 bit, only the lower part needs to be written. The most-significant bit is then taken as sign-bit for the number.

| | |
|---------------|--------------------------------------|
| Index | 3475h |
| Name | DOUTx.PARAM |
| Object code | Array |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0x4 |
| Default value | 0x4 |
| Subindex | 1 to 2 |
| Description | DOUTx.PARAM low 32 bits, x = 1 .. 2 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0 |
| Subindex | 3 to 42 |
| Description | DOUTx.PARAM high 32 bits, x = 1 .. 2 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0 |

21.3.50 Object 3496h: Fieldbus synchronization parameters

This set of objects is used to set or read parameters for the fieldbus synchronization used in the interpolated position mode (7) and the cyclic-modes 8 etc. The synchronization between a fieldbus master and the AKD PDMM is similar in all the supported fieldbus systems.

The AKD PDMM internal 16[kHz] interrupt function is responsible for calling the PLL function. This PLL function is called once per fieldbus cycle (set by object 60C2 sub 1 and 2). If the fieldbus sample period is for example 1[ms], the PLL code is called every 16th time of the 16[kHz] IRQ of the AKD PDMM.

Once in a fieldbus sample the SYNC-telegram must arrive, which resets a PLL counter in the Drive. After some time the already mentioned PLL function is called and reads back the time from that PLL counter.

Depending on the measured time the PLL function extends (in case that the measured time is too low) or lowers (in case that the measured time is too high) the sample time of the upcoming 16[kHz] tasks for one fieldbus sample by a selectable value (object 3496 sub 4) in order to move the PLL function closer to the expected distance (object 3496 sub 1).

Beside the objects mentioned here the parameter FBUS.SAMPLEPERIOD is important, which is set by object 60C2 sub 1 and 2. This setting is required in order to share the fieldbus sample time with the slave. This information is e.g. needed for being able to call the AKD PDMM internal PLL function once per fieldbus sample.

| | |
|---------------|---------------------------------|
| Index | 3496h |
| Name | FBUS synchronization parameters |
| Object code | Array |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 0x4 |
| Default value | 0x4 |
| Subindex | 1 |
| Description | FBUS.SYNCDIST |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 320000 [ns] |

Sub 1 is the expected time distance in nano seconds between clearing the PLL counter and calling the PLL function.

| | |
|---------------|--------------|
| Subindex | 2 |
| Description | FBUS.SYNCACT |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 320000 [ns] |

Sub 2 is the actual time distance in nano seconds between clearing the PLL counter and calling the PLL function.

| | |
|---------------|--------------|
| Subindex | 3 |
| Description | FBUS.SYNCWND |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 70000 [ns] |

Sub 3 is a window, which is used in order to consider the drive as being synchronized. The AKD PDMM is considered as synchronized in the following case:

$$FBUS.SYNCDIST - FBUS.SYNCWND < FBUS.SYNCACT < FBUS.SYNCDIST + FBUS.SYNCWND$$

| | |
|---------------|---------------|
| Subindex | 4 |
| Description | FBUS.COMPTIME |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 150 [ns] |

Sub 4 value indicates the time, which is used for extending or lowering the sample rate of the AKD PDMM internal 16[kHz] IRQ, which is responsible for calling the PLL function. The default sample time is $32 * 1/16$ [kHz] = 2[ms].

The sample time of the AKD PDMM high prior interrupt is determined by $62.5[\mu\text{s}] - FBUS.COMPTIME$ if $FBUS.SYNCACT > FBUS.SYNCDIST$.

The sample time of the AKD PDMM high prior interrupt is determined by $62.5[\mu\text{s}] + FBUS.COMPTIME$ if $FBUS.SYNCACT < FBUS.SYNCDIST$.

21.3.51 Object 6040h: Control word (DS402)

The control commands are built up from the logical combination of the bits in the control word and external signals (e.g enable output stage). The definitions of the bits are shown below:

| | |
|---------------|--------------|
| Index | 6040h |
| Name | control word |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Access | R/W |
| PDO mapping | possible |
| Unit | — |
| Value range | 0 to 65535 |
| EEPROM | no |
| Default value | 0 |

Bit assignment in control word

| Bit | Name | Bit | Name |
|-----|---|-----|-----------------------|
| 0 | Switch on | 8 | Pause/halt |
| 1 | Disable Voltage | 9 | reserved |
| 2 | Quick Stop | 10 | reserved |
| 3 | Enable Operation | 11 | reserved |
| 4 | Operation mode specific | 12 | reserved |
| 5 | Operation mode specific | 13 | Manufacturer-specific |
| 6 | Operation mode specific | 14 | Manufacturer-specific |
| 7 | Reset Fault (only effective for faults) | 15 | Manufacturer-specific |

Commands in the control word

| Command | Bit 7 Fault Reset | Bit 3 Enable Operation | Bit 2 Quick Stop | Bit 1 Disable Voltage | Bit 0 Switch on | Transitions |
|------------------------|-------------------------|------------------------------|------------------------|-----------------------------|-----------------------|--------------|
| Shutdown | X | X | 1 | 1 | 0 | 2, 6, 8 |
| Switch on | X | X | 1 | 1 | 1 | 3 |
| Disable Voltage | X | X | X | 0 | X | 7, 9, 10, 12 |
| Quick Stop | X | X | 0 | 1 | X | 7, 10, 11 |
| Disable Oper- ation | X | 0 | 1 | 1 | 1 | 5 |
| Enable Operation | X | 1 | 1 | 1 | 1 | 4, 16 |
| Fault Reset | 1 | X | X | X | X | 15 |

Bits marked by an X are irrelevant.

Mode-dependent bits in the control word

The following table shows the mode-dependent bits in the control word. Only manufacturer-specific modes are supported at present. The individual modes are set by Object 6060_n Modes of operation.

| Operation mode | No. | Bit 4 | Bit 5 | Bit 6 |
|---------------------------------|-----|------------------------|------------------------|-------------------|
| Profile Position Mode (pp) | 01h | new_setpoint | change_set_immediately | absolute/relative |
| Profile Velocity Mode (pv) | 03h | reserved | reserved | reserved |
| Profile Torque Mode (tq) | 04h | reserved | reserved | reserved |
| Homing Mode (hm) | 06h | homing_operation_start | reserved | reserved |
| Interpolated Position Mode (ip) | 07h | Enable Interpolation | reserved | reserved |
| Cyclic sync position Mode (csp) | 08h | reserved | reserved | reserved |

Description of the remaining bits in the control word

The remaining bits in the control word are described below.

Bit 8 Pause If Bit 8 is set, then the drive halts (pauses) in all modes. The setpoints (speed for homing or jogging, motion task number, setpoints for digital mode) for the individual modes are retained.

Bit 9,10 These bits are reserved for the drive profile (DS402).

Bit 13, 14, 15 These bits are manufacturer-specific, and reserved at present.

21.3.52 Object 6041h: Status word (DS402)

The momentary state of the state machine can be read out with the aid of the status word.

| | |
|---------------|-------------|
| Index | 6041h |
| Name | Status word |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Access | R/W |
| PDO mapping | possible |
| Unit | — |
| Value range | 0 to 65535 |
| EEPROM | yes |
| Default value | 0 |

Bit assignment in the status word

| Bit | Name | Bit | Name |
|-----|--------------------|-----|------------------------------------|
| 0 | Ready to switch on | 8 | STO – Safe Torque Off |
| 1 | Switched on | 9 | Remote |
| 2 | Operation enabled | 10 | Target reached |
| 3 | Fault | 11 | Internal limit active |
| 4 | Voltage enabled | 12 | Operation mode specific (reserved) |
| 5 | Quick stop | 13 | Operation mode specific (reserved) |

| | | | |
|---|--------------------|----|----------------------------------|
| 6 | Switch on disabled | 14 | Manufacturer-specific (reserved) |
| 7 | Warning | 15 | Manufacturer-specific (reserved) |

States of the state machine

| State | Bit 6 switch on disabled | Bit 5 quick stop | Bit 3 fault | Bit 2 operation enabled | Bit 1 switched on | Bit 0 ready to switch on |
|------------------------|--------------------------------|---------------------|----------------|-------------------------------|-------------------------|-----------------------------------|
| Not ready to switch on | 0 | X | 0 | 0 | 0 | 0 |
| Switch on disabled | 1 | X | 0 | 0 | 0 | 0 |
| Ready to switch on | 0 | 1 | 0 | 0 | 0 | 1 |
| Switched on | 0 | 1 | 0 | 0 | 1 | 1 |
| Operation enabled | 0 | 1 | 0 | 1 | 1 | 1 |
| Fault | 0 | X | 1 | 0 | 0 | 0 |
| Fault reaction active | 0 | X | 1 | 1 | 1 | 1 |
| Quick stop active | 0 | 0 | 0 | 1 | 1 | 1 |

Bits marked by X are irrelevant

Description of the remaining bits in the status word

Bit 4: voltage_enabled The DC-link voltage is present if this bit is set.

Bit 7: warning There are several possible reasons for Bit 7 being set and this warning being produced. The reason of a warning can be seen by the Error code of the Emergency message, which is sent on the bus caused by this warning.

Bit 9: The remote-bit is set by the telnet command FBUS.REMOTE. The default state is 1 indicating that the power stage shall be only controlled by the DS402 control word. For special actions via telnet like tuning or commutation finding, FBUS.REMOTE shall be set to 0 via telnet to inform the fieldbus master.

Bit 10: target_reached This is set when the drive has reached the target position. In profile position mode the position window is set via MT.TPOSWND, in homing mode via HOME.TPOSWND.

Bit 11: internal_limit_active This bit specifies that a movement was or is limited. In different modes, different warnings cause the bit to be set. The following assignments exist:

| Mode of operation | Warnings which set Bit 11 |
|-------------------|------------------------------|
| all | n04, n06, n07, n10, n11, n14 |
| 0x1 (PP), 0x88 | n03, n08, n09, n20 |

Mode-dependent bits in the status word

The following table shows the mode-dependent bits in the status word. The individual modes are set by " Object 6060h: Modes of Operation (DS402)" (→ p. 1053).

| Operation mode | No. | Bit 12 | Bit 13 |
|---------------------------------|-----|--|-----------------|
| Profile Position Mode (pp) | 01h | setpoint acknowledge | following error |
| Homing Mode (hm) | 06h | homing attained | homing error |
| Interpolated Position Mode (ip) | 07h | ip mode active | following error |
| Cyclic sync position Mode (csp) | 08h | This bit stays on 1 as long as the drive is following the position set-points. | following error |

21.3.53 Object 605Ah: Quick stop option code (DS402)

This object defines the action, which is taken as quick stop function.

| | |
|---------------|------------------------|
| Index | 605Ah |
| Name | Quick stop option code |
| Object code | VAR |
| Data type | INTEGER16 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | 1, 2, 5, 6 |
| Default value | 2 |

Supported codes:

| Bit | Description |
|-----|--|
| 1 | Slow down on slow down ramp and transit into Switch On Disabled |
| 2 | Slow down on quick stop ramp and transit into Switch On Disabled |
| 5 | Slow down on slow down ramp and stay in Quick Stop Active |
| 6 | Slow down on quick stop ramp and stay in Quick Stop Active |

21.3.54 Object 6060h: Modes of Operation (DS402)

This object is used to set the mode, which can be read out by Object 6061h. Two types of operating mode are used:

- manufacturer-specific operating modes
- operating modes as per CANopen drive profile DS402

These operating modes are defined in the CANopen drive profile DS402. After the mode has been changed, the corresponding setpoint must be set once more (for instance, the homing velocity in the mode homing_setpoint). If the position or jogging mode is stored, then the Homing mode is set after a RESET of the drive.

NOTE

An operating mode only becomes valid when it can be read by Object 6061h.



! WARNING

Automatic Start

Risk of death or serious injury for humans working in the machine. The drive could move unexpectedly. When the drive is enabled, a mode change is only permissible at zero speed.

- Never change the mode while the motor is running!
- Set the speed setpoint to 0 before changing over.

| | |
|---------------|------------------------------|
| Index | 6060h |
| Name | mode of operation |
| Object code | VAR |
| Data type | INTEGER8 |
| Category | mandatory |
| Access | R/W |
| PDO mapping | possible |
| Value range | -3, -2, -1, 1, 3, 4, 6, 7, 8 |
| Default value | — |

Supported modes (negative values are manufacturer specific modes):

| Value (hex) | Mode |
|-------------|----------------------------------|
| -3 | Electronic gearing mode |
| -2 | Analog velocity mode |
| -1 | Analog current mode |
| 1 | Profile position mode |
| 3 | Profile velocity mode |
| 4 | Profile torque mode |
| 6 | Homing mode |
| 7 | Interpolated position mode |
| 8 | Cyclic synchronous position mode |

21.3.55 Object 6061h: Modes of Operation Display (DS402)

This object can be used to read the mode that is set by Object 6060h. An operating mode only becomes valid when it can be read by Object 6061h (see also Object 6060h).

| | |
|---------------|------------------------------|
| Index | 6061h |
| Name | mode of operation display |
| Object code | VAR |
| Data type | INTEGER8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | possible |
| Value range | -3, -2, -1, 1, 3, 4, 6, 7, 8 |
| Default value | — |

21.3.56 Object 6063h: position actual value* (DS402)

The object position actual value provides the momentary actual position in increments. The resolution is defined with Object 608F as power-of-two number.

| | |
|---------------|---|
| Index | 6063h |
| Name | position actual value |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | pc, pp |
| Access | R/W |
| PDO mapping | possible |
| Unit | increments (1 turn = $2^{FB1.PSCALE}$) |
| Value range | (-2^{31}) to $(2^{31}-1)$ |
| Default value | 2^{20} |
| EEPROM | no |

21.3.57 Object 6064h: position actual value (DS402)

The object position actual value provides the actual position. The resolution can be altered by the gearing factors of the position controller (Object 6091/6092).

| | |
|-------------|------------------------------|
| Index | 6064h |
| Name | position actual value, PL.FB |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | pp, csp |
| Access | R/W |
| PDO mapping | possible |

| | |
|---------------|-----------------------------|
| Unit | position units |
| Value range | (-2^{31}) to $(2^{31}-1)$ |
| Default value | — |
| EEPROM | no |

21.3.58 Object 6065h: Following error window

The following error window defines a range of tolerated position values symmetrically to the position demand value. A following error might occur when a drive is blocked, unreachable profile velocity occurs, or at wrong closed loop coefficients. If the value of the following error window is 0, the following control is switched off.

| | |
|---------------|------------------------|
| Index | 6065h |
| Name | Following error window |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0 |

21.3.59 Object 606Ch: Velocity actual value (DS402)

The object velocity actual value represents the actual speed.

| | |
|---------------|---|
| Index | 606Ch |
| Name | velocity actual value, VL.FB |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | pv |
| Access | R/O |
| PDO mapping | possible |
| Unit | velocity units (SDO is in user units and the PDO is in RPM) |
| Value range | (-2^{31}) to $(2^{31}-1)$ |
| Default value | — |
| Float scaling | 1000:1 |
| EEPROM | no |

21.3.60 Object 6071h: Target torque (DS402)

This parameter is the input value for the torque controller in profile torque mode and the value is given per thousand (1‰) of rated torque.

| | |
|---------------|---|
| Index | 6071h |
| Name | Target torque |
| Object code | VAR |
| Data type | INTEGER16 |
| Category | conditional; mandatory, if tq supported |
| Access | R/W |
| PDO mapping | possible |
| Value range | INTEGER16 |
| Default value | 0 |

21.3.61 Object 6073h: Max current (DS402)

This value represents the maximum permissible torque creating current in the motor and is given per thousand (1‰) of rated current.

| | |
|---------------|--------------|
| Index | 6073h |
| Name | Max current |
| Object code | VAR |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED16 |
| Default value | 0 |

21.3.62 Object 6077h: Torque actual value (DS402)

The torque actual value corresponds to the instantaneous torque in the drive motor. The value is given per thousand (1‰) of rated torque. This object reflects the actual current (DRV.ICONT).

| | |
|---------------|---------------------|
| Index | 6077h |
| Name | Torque actual value |
| Object code | VAR |
| Data type | INTEGER16 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER16 |
| Default value | 0 |

21.3.63 Object 607Ah: Target position (DS402)

The object target position defines the target position for the drive. The target position is interpreted as a relative distance or an absolute position, depending on Bit 6 of the control word. The type of relative movement can be further defined by the manufacturer-specific parameter 35B9h Subindex 0. Other properties like following motion tasks can be set with this object as well. The mechanical resolution is set via the scaling objects 6091h and 6092h.

| | |
|---------------|-------------------------------|
| Index | 607Ah |
| Name | target position, MT.P |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | pp, csp |
| Access | R/W |
| PDO mapping | possible |
| Unit | user-defined |
| Value range | $-(2^{31}-1)$ to $(2^{31}-1)$ |
| Default value | — |

21.3.64 Object 607Ch: Homing offset (DS402)

The reference offset (home offset) is the difference between the zero position for the application and the zero point of the machine. All subsequent absolute motion tasks take account of the reference offset.

| | |
|---------------|-----------------------------|
| Index | 607Ch |
| Name | home offset, HOME.P |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | hm |
| Access | R/W |
| PDO mapping | not possible |
| Unit | user-defined |
| Value range | (-2^{31}) to $(2^{31}-1)$ |
| Default value | 0 |

21.3.65 Object 607Dh: Software position limit (DS402)

Software position limit contains the sub-parameters min position limit and max position limit. New target positions are checked against these limits. The limits are relative to the machine home position, which is the result of homing (including the home offset (Object 607Ch)). As default the software position limits are switched off. Changed values must be saved and the drive must be restarted to take enable the new the software limits.

| | |
|---------------|--------------------------------------|
| Index | 607Dh |
| Name | Software position limit, SWLS.LIMIT0 |
| Object code | ARRAY |
| Data type | INTEGER32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | 2 |
| Subindex | 1 |
| Description | min position limit 1, SWLS.LIMIT0 |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | 0 (switched off) |
| Subindex | 2 |
| Description | Min Position Limit 2, SWLS.LIMIT1 |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | 0 (switched off) |

21.3.66 Object 6081h: Profile velocity (DS402)

The profile velocity is the final velocity that should be reached after the acceleration phase of a motion task.

| | |
|---------------|------------------------|
| Index | 6081h |
| Name | profile velocity, MT.V |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Mode | pp |
| Access | R/W |
| PDO mapping | possible |
| Unit | speed units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | 10 |

21.3.67 Object 6083h: Profile acceleration (DS402)

The acceleration ramp (profile acceleration) is given in units that are defined by the user (position units per s²). The position units are scaled via the objects 6091 and 6092. This object is connected to the AKD PDMM-parameter DRV.ACC in the Profile Velocity Mode and to the motion task parameter MT.ACC in all other modes.

| | |
|---------------|---|
| Index | 6083h |
| Name | profile acceleration, MT.ACC (DRV.ACC in Profile Velocity Mode) |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Mode | pp, pv |
| Access | R/W |
| PDO mapping | possible |
| Unit | acceleration units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | 0 |

21.3.68 Object 6084h: Profile deceleration (DS402)

The braking/deceleration ramp is handled in the same way as the acceleration ramp (" Object 6083h: Profile acceleration (DS402)" (→ p. 1059)).

| | |
|-------------|---|
| Index | 6084h |
| Name | profile deceleration, MT.DEC (DRV.DEC in Profile Velocity Mode) |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Mode | pp, pv |
| Access | R/W |
| PDO mapping | possible |

| | |
|---------------|--------------------|
| Unit | deceleration units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | 0 |

21.3.69 Object 6087h Torque slope (DS402)

This object defines the rate of change of torque. The value is given in units of per thousand of rated torque per second. The minimum rate settable for the AKD PDMM is equivalent to the value of DRV.ICONT (= 1000 per mille) per ~ 420 milliseconds equivalent to a value of ~ 2385 per mille / second.

| | |
|---------------|-------------------------------|
| Index | 6087h |
| Name | Torque slope |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | mandatory, if tq is supported |
| Access | R/W |
| PDO mapping | possible |
| Value range | UNSIGNED32 |
| Default value | - |

21.3.70 Object 608Fh: Position encoder resolution (DS402)

The position encoder resolution defines the ratio of encoder increments per motor revolution on the CANopen end. Encoder increments are set either directly by subindex 1 (only powers of 2 available) or implicit by writing to the parameter FB1.PSCALE.

| | |
|---------------|-----------------------------|
| Index | 608Fh |
| Name | Position encoder resolution |
| Object Code | ARRAY |
| Data Type | UNSIGNED 32 |
| Category | optional |
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 2 |
| Default Value | 2 |
| Subindex | 1 |
| Name | Encoder increments |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 2 ²⁰ |
| Subindex | 2 |

| | |
|---------------|-------------------|
| Name | Motor revolutions |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 1 |

21.3.71 Object 6091h: Gear Ratio (DS402)

The gear ratio defines the ratio of feed in position units per driving shaft revolutions. This includes the gear if present.

gear ratio = motor shaft revolutions / driving shaft revolutions

| | |
|---------------|-----------------------------|
| Index | 6091h |
| Name | Gear Ratio |
| Object Code | ARRAY |
| Data Type | UNSIGNED 32 |
| Category | optional |
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 2 |
| Default Value | 2 |
| Subindex | 1 |
| Name | Motor revolution |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 1 |
| Subindex | 2 |
| Name | Shaft revolutions |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 1 |

21.3.72 Object 6092h: Feed constant (DS402)

The feed constant defines the ratio of feed in position units per driving shaft revolutions. This includes the gear if present.

| | |
|---------------|-----------------------------|
| Index | 6092h |
| Name | Feed constant |
| Object Code | ARRAY |
| Data Type | UNSIGNED 32 |
| Category | optional |
| Subindex | 0 |
| Name | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO Mapping | not possible |
| Value Range | 2 |
| Default Value | 2 |
| Subindex | 1 |
| Name | Feed |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 1 |
| Subindex | 2 |
| Name | Shaft revolutions |
| Category | mandatory |
| Access | R/W |
| PDO Mapping | not possible |
| Value Range | UNSIGNED 32 |
| Default Value | 1 |

21.3.73 Object 6098h: Homing method (DS402)

| | |
|---------------|------------------------------------|
| Index | 6098h |
| Name | homing method, HOME.MODE, HOME.DIR |
| Object code | VAR |
| Data type | INTEGER8 |
| Mode | hm |
| Access | R/W |
| PDO mapping | not possible |
| Unit | position units |
| Value range | -128 to 127 |
| Default value | 0 |

Description of the homing methods

Choosing a homing method by writing a value to homing method (Object 6098h) will clearly establish:

- the homing signal (P-Stop, N-Stop, reference switch)
- the direction of actuation

and where appropriate

- the position of the index pulse.

The reference position is give by the reference offset (Object 607Ch).

A detailed description of the types of homing movement can be found in the description of [EWV¹](#).

The following homing methods are supported:

| Method as per DS402 | Brief description: Homing | command |
|---------------------|---|---------------------------------------|
| -128 to -8 | reserved | — |
| -7 | find home input in negative direction, then find next feedback zero | HOME.MODE=18, HOME.DIR=0 |
| -6 | find home input in positive direction, then find next feedback zero | HOME.MODE=18, HOME.DIR = 1 |
| -5 | set reference point at actual position and store value in NVRAM | HOME.MODE=17, HOME.DIR not changed |
| -4 | find reference switch with fast velocity (6099h sub1) and home on reference switch with low velocity (6099h sub 2), positive count direction | HOME.MODE=16, HOME.DIR=0 |
| -3 | find reference switch with fast velocity (6099h sub 1) and home on reference switch with low velocity (6099h sub 2), negative count direction | HOME.MODE=16, HOME.DIR=0 |
| -2 to -1 | reserved | — |
| 0 | reserved | — |
| 1 | homing to negative limit switch, with zeroing, negative count direction | HOME.MODE=2, HOME.DIR=0 |

¹Embedded Workbench Views

| Method as per DS402 | Brief description: Homing | command |
|---------------------|---|------------------------------|
| 2 | homing to positive limit switch, with zeroing, positive count direction | HOME.MODE=2, HOME.DIR=1 |
| 3 to 7 | not supported | — |
| 8 | homing to reference switch, with zeroing, positive count direction | HOME.MODE=5, HOME.DIR=1 |
| 9 to 11 | not supported | — |
| 12 | homing to reference switch, with zeroing, negative count direction | HOME.MODE=5, HOME.DIR=0 |
| 13 to 14 | not supported | — |
| 15 to 16 | reserved | — |
| 17 | homing to negative limit switch, without zeroing, negative count direction | HOME.MODE=1, HOME.DIR=0 |
| 18 | homing to negative limit switch, without zeroing, positive count direction | HOME.MODE=1, HOME.DIR=1 |
| 19 to 23 | not supported | — |
| 24 | homing to reference switch, without zeroing, positive count direction | HOME.MODE=4, HOME.DIR=1 |
| 25 to 27 | not supported | — |
| 28 | homing to reference switch, without zeroing, negative count direction | HOME.MODE=4, HOME.DIR=0 |
| 29 to 30 | not supported | — |
| 31 to 32 | reserved | — |
| 33 | homing within a single turn, negative count direction. If the feedback has an index pulse, HOME.MODE 11 will be used. | HOME.MODE=7,11 HOME.DIR=0 |
| 34 | homing within a single turn, positive count direction. If the feedback has an index pulse, HOME.MODE 11 will be used. | HOME.MODE=7,11 HOME.DIR=1 |
| 35 | set reference point at present position | HOME.MODE=0, HOME.DIR=0 |
| 36 to 127 | reserved | — |

21.3.74 Object 6099h: Homing speeds (DS402)

| | |
|-------------|--|
| Index | 6099h |
| Name | homing speeds |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Subindex | 1 |
| Description | speed during search for switch, HOME.V |
| Mode | hm |
| Access | R/W |

| | |
|---------------|---|
| PDO mapping | not possible |
| Unit | velocity units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | equivalent 60 rpm |
| Subindex | 2 |
| Description | speed during search for zero, HOME.FEEDRATE |
| Mode | hm |
| Access | R/W |
| PDO mapping | not possible |
| Unit | velocity units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | 50% of Object 6099 sub 1 |

21.3.75 Object 609Ah: Homing acceleration (DS402)

| | |
|---------------|---------------------|
| Index | 609Ah |
| Name | homing acceleration |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Mode | hm |
| Access | R/W |
| PDO mapping | not possible |
| Unit | acceleration units |
| Value range | 0 to $(2^{32}-1)$ |
| Default value | 0 |

21.3.76 Object 60B1h: Velocity Offset

This object provides the offset of the velocity value in cyclic synchronous position mode. It is scaled via the object 204Ch.

| | |
|---------------|-----------------|
| Index | 60B1h |
| Name | Velocity Offset |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/W |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Default value | 0 |

21.3.77 Object 60B2h: Torque Offset

This object (IL.BUSFF) provides the offset of the commanded torque from a bus network connected to the drive. Scaling is 1/1000 of rated torque.

| | |
|---------------|---------------|
| Index | 60B2h |
| Name | Torque Offset |
| Object code | VAR |
| Data type | INTEGER16 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER16 |
| Default value | 0 |

21.3.78 Object 60B8h: Touch probe function

This object indicates the configured function of the touch probe.

| | |
|---------------|----------------------|
| Index | 60B8h |
| Name | Touch probe function |
| Object code | Variable |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | R/W |
| PDO Mapping | yes |
| Value range | UNSIGNED16 |
| Default value | 0 |

Definition of the possible functions:

| Bit | Value | Meaning |
|--------|-------|--|
| 0 | 0 | Switch off touch probe 1 |
| | 1 | Enable touch probe 1 |
| 1 | 0 | Trigger first event |
| | 1 | Continuous |
| 3, 2 | 00b* | Trigger with touch probe 1 input |
| | 01b | Trigger with zero impulse signal or position encoder |
| | 10b | Touch probe source as defined in object 60D0h, sub-index 01h |
| | 11b | reserved |
| 4 | 0 | Switch off sampling at positive edge of touch probe 1 |
| | 1 | Enable sampling at positive edge of touch probe 1 |
| 5 | 0 | Switch off sampling at negative edge of touch probe 1 |
| | 1 | Enable sampling at negative edge of touch probe 1 |
| 6, 7 | - | User-defined (e.g. for testing) |
| 8 | 0 | Switch off touch probe 2 |
| | 1 | Enable touch probe 2 |
| 9 | 0 | Trigger first event |
| | 1 | continuous |
| 11, 10 | 00b | Trigger with touch probe 2 input |
| | 01b | Trigger with zero impulse signal or position encoder |
| | 10b | Touch probe source as defined in object 60D0h, sub-index 02h |
| | 11b | reserved |
| 12 | 0 | Switch off sampling at positive edge of touch probe 2 |
| | 1 | Enable sampling at positive edge of touch probe 2 |
| 13 | 0 | Switch off sampling at negative edge of touch probe 2 |
| | 1 | Enable sampling at negative edge of touch probe 2 |
| 14, 15 | - | User-defined (e.g. for testing) |

* b = binary

If both edges are selected at the same time (bit 4=1 and bit 5=1 for probe 1 or bit 12=1 and bit 13=1 for probe 2), the first edge (positive or negative) triggers the probe function. The position, latched at this edge, is taken over for both edges (positive and negative).

21.3.79 Object 60B9h: Touch probe status

This object indicates the status of the touch probe.

| | |
|---------------|--------------------|
| Index | 60B9h |
| Name | Touch probe status |
| Object code | Variable |
| Data type | UNSIGNED16 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | UNSIGNED16 |
| Default value | 0 |

Definition of the status:

| Bit | Value | Meaning |
|----------|-------|---|
| 0 | 0 | Touch probe 1 is switched off |
| | 1 | Touch probe 1 is enabled |
| 1 | 0 | Touch probe 1 no positive edge value stored |
| | 1 | Touch probe 1 positive edge position stored |
| 2 | 0 | Touch probe 1 no negative edge value stored |
| | 1 | Touch probe 1 negative edge position stored |
| 3 to 5 | 0 | reserved |
| 6, 7 | - | User-defined (e.g. for testing) |
| 8 | 0 | Touch probe 2 is switched off |
| | 1 | Touch probe 2 is enabled |
| 9 | 0 | Touch probe 2 no positive edge value stored |
| | 1 | Touch probe 2 positive edge position stored |
| 10 | 0 | Touch probe 2 no negative edge value stored |
| | 1 | Touch probe2 negative edge position stored |
| 11 to 13 | 0 | reserved |
| 14, 15 | - | User-defined (e.g. for testing) |

21.3.80 Object 60BAh: Touch probe 1 positive edge

This object provides the position value of the touch probe 1 at positive edge.

| | |
|---------------|-----------------------------|
| Index | 60BAh |
| Name | Touch probe 1 positive edge |
| Object code | Variable |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | INTEGER32 |
| Default value | no |

21.3.81 Object 60BBh: Touch probe 1 negative edge

This object provides the position value of the touch probe 1 at negative edge.

| | |
|---------------|-----------------------------|
| Index | 60BBh |
| Name | Touch probe 1 negative edge |
| Object code | Variable |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | INTEGER32 |
| Default value | no |

21.3.82 Object 60BCh: Touch probe 2 positive edge

This object provides the position value of the touch probe 2 at positive edge.

| | |
|---------------|-----------------------------|
| Index | 60BCh |
| Name | Touch probe 2 positive edge |
| Object code | Variable |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | INTEGER32 |
| Default value | no |

21.3.83 Object 60BDh: Touch probe 2 negative edge

This object provides the position value of the touch probe 2 at negative edge.

| | |
|---------------|-----------------------------|
| Index | 60BDh |
| Name | Touch probe 2 negative edge |
| Object code | Variable |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | INTEGER32 |
| Default value | no |

21.3.84 Object 60C0h: Interpolation sub mode select

In the AKD PDMM, linear interpolation between position setpoints is supported.

| | |
|---------------|-------------------------------|
| Index | 60C0h |
| Name | Interpolation sub mode select |
| Object code | VAR |
| Data type | INTEGER16 |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | 0 |
| Default value | 0 |

Value description

| Value(decimal) | Description |
|----------------|--|
| 0 | Linear interpolation with a constant time. |

21.3.85 Object 60C1h: Interpolation data record

In the AKD PDMM, a single setpoint (target position, Subindex 1) is supported for the linear interpolation. After the last item of an interpolation data record is written to the devices input buffer, the buffer pointer is automatically incremented to the next buffer.

| | |
|---------------|--|
| Index | 60C1h |
| Name | Interpolation data record |
| Object code | ARRAY |
| Data type | INTEGER32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Data type | UNSIGNED8 |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 1 |
| Default value | 1 |
| Subindex | 1 |
| Description | Interpolation target position in counts, the first parameter of interpolation function |
| Category | mandatory |
| Access | R/W |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Default value | no |

NOTE

A set-point value of the Interpolation data record is only taken, if beside the state machine state "Operation Enable" also the bit 4 of the DS402 controlword (Enable Interpolation, see "Object 6040h: Control word (DS402)" (→ p. 1048)) is set.

21.3.86 Object 60C2h: Interpolation time period

The interpolation time period is used for the PLL (phase locked loop) synchronized position modes. The unit (subindex 1) of the time is given in 10⁹interpolation time index seconds.

Only multiples of 1 ms are allowed. The two values define the internal AKD PDMM parameter FBUS.SAMPLEPERIOD (given in multiples of 62.5 Mikroseconds). Both values must be written to set a new interpolation time period. FBUS.SAMPLEPERIOD will only be updated then.

| | |
|---------------|--|
| Index | 60C2h |
| Name | Interpolation time period |
| Object code | RECORD |
| Data type | Interpolation time period record (0080h) |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported, FBUS.SAMPLEPERIOD |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | 2 |
| Subindex | 1 |
| Description | Interpolation time units |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | 2 |
| Subindex | 2 |
| Description | Interpolation time index |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER8 |
| Default value | -3 |

21.3.87 Object 60C4h: Interpolation data configuration

In the AKD PDMM, for linear interpolation, only the value 1 in Subindex 5 is possible. For cubic interpolation, Subindex 1 and Subindex 2 is possible for manage the buffer of cubic interpolation.

| | |
|---------------|---|
| Index | 60C4h |
| Name | Interpolation data configuration |
| Object code | RECORD |
| Data type | Interpolation data configuration record (0081h) |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 6 |
| Default value | 6 |
| Subindex | 1 |
| Description | Maximum buffer size |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 10 |
| Subindex | 2 |
| Description | Actual buffer size |
| Category | mandatory |
| Access | R/O |
| PDO mapping | possible |
| Value range | 0 to 9 |
| Default value | 9 |
| Subindex | 3 |
| Description | Buffer organization |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGND8 |
| Default value | 0 |

| | |
|---------------|---------------------|
| Subindex | 4 |
| Description | Buffer position |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED16 |
| Default value | 0 |
| Subindex | 5 |
| Description | Size of data record |
| Category | mandatory |
| Access | W |
| PDO mapping | not possible |
| Value range | 1 to 254 |
| Default value | 1 |
| Subindex | 6 |
| Description | Buffer clear |
| Category | mandatory |
| Access | W |
| PDO mapping | not possible |
| Value range | UNSIGNED8 |
| Default value | 0 |

21.3.88 Object 60D0h: Touch probe source

This object provides the source of the touch probe function, when the dedicated bits 2/3 or 10/11 of the touch probe function (object 60B8h) are set accordingly.

| | |
|---------------|-----------------------------|
| Index | 60D0h |
| Name | Touch probe source |
| Object code | Array |
| Data type | Integer 16 |
| Category | optional |
| Subindex | 0 |
| Description | Highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 2 |
| Default value | 2 |
| Subindex | 1 |
| Description | Touch probe 1 source |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | -11 to -1, 1 to 5 |
| Default value | 1 |
| Subindex | 2 |
| Description | Touch probe 2 source |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | -11 to -1, 1 to 5 |
| Default value | 1 |

Value description:

| Value | Description | Value | Description |
|-----------|---|-------|---------------------|
| 1 | Touch Probe 1 Input | 3 | Touch Probe 3 Input |
| 2 | Touch Probe 2 Input | 4 | Touch Probe 4 Input |
| -1 to -11 | AKD PDMM Input related to CAPx.TRIGGER 0 to 10 | | |

21.3.89 Object 60E0h: Positive Torque Limit Value

The object gives the configured maximum motor torque in positive direction. The value is given per thousand (1 ‰) of rated torque.

| | |
|---------------|--|
| Index | 60E0h |
| Name | Positive Torque Limit Value |
| Object code | Variable |
| Data type | UIINTEGER16 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | UIINTEGER16 (limited by DRV.IPEAK and MOTOR.IPEAK) |
| Default value | 0 |

21.3.90 Object 60E1h: Negative Torque Limit Value

The object gives the configured maximum motor torque in negative direction. The value is given per thousand (1 ‰) of rated torque.

| | |
|---------------|--|
| Index | 60E1h |
| Name | Negative Torque Limit Value |
| Object code | Variable |
| Data type | UIINTEGER16 |
| Category | optional |
| Access | R/O |
| PDO Mapping | yes |
| Value range | UIINTEGER16 (limited by DRV.IPEAK and MOTOR.IPEAK) |
| Default value | 0 |

21.3.91 Object 60E4h: Additional position actual value

This object provides the additional position actual values. The values are given in user-defined position units. The value is calculated analog to the calculation for the actual position 6064h via object 6091h and 6092h, but for this with the factors given by the objects 60E8h, 60E9h, 60EDh and 60EEh.

| | |
|---------------|--------------------------------------|
| Index | 60E4h |
| Name | Additional position actual value |
| Object code | ARRAY |
| Data type | INTEGER32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 |
| Description | 1st additional position actual value |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | 1 |
| Subindex | 2 |
| Description | reserved |
| Category | |
| Access | |
| PDO mapping | |
| Value range | |
| Default value | |
| Subindex | 3 |
| Description | 3rd additional position actual value |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | INTEGER32 |
| Default value | 0 |

21.3.92 Object 60E8h: Additional gear ratio – motor shaft revolutions

This object provides the motor shaft revolutions for the additional gear ratio calculation. This object shall be used with the corresponding subindex of the object 60EDh (driving shaft revolutions for the additional gear ratio calculation). The value of the object 60E4h is calculated analog to the gear ration calculation for the actual position 6064h via object 6091h.

| | |
|---------------|---|
| Index | 60E8h |
| Name | Additional gear ratio – motor shaft revolutions |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 |
| Description | 1st additional gear ratio - motor shaft |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |
| Subindex | 2 |
| Description | reserved |
| Category | |
| Access | |
| PDO mapping | |
| Value range | |
| Default value | |
| Subindex | 3 |
| Description | 3rd additional gear ratio - motor shaft |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |

21.3.93 Object 60E9h: Additional feed constant – feed

This object provides the feed for the additional feed constant calculation. This object shall be used with the corresponding subindex of the object 60EEh (driving shaft revolutions for the additional feed constant calculation). The value of the object 60E4h is calculated analog to the feed constant calculation for the actual position 6064h via object 6092h.

| | |
|---------------|-------------------------------------|
| Index | 60E9h |
| Name | Additional feed constant – feed |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 |
| Description | 1st additional feed constant – feed |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |
| Subindex | 2 |
| Description | reserved |
| Category | |
| Access | |
| PDO mapping | |
| Value range | |
| Default value | |
| Subindex | 3 |
| Description | 3rd additional feed constant – feed |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |

21.3.94 Object 60EDh: Additional gear ratio – driving shaft revolutions

This object provides the driving shaft revolutions for the additional gear ratio calculation. This object shall be used with the corresponding subindex of the object 60E8h (motor shaft revolutions for the additional gear ratio calculation). The value of the object 60E4h is calculated analog to the gear ration calculation for the actual position 6064h via object 6091h.

| | |
|---------------|---|
| Index | 60EDh |
| Name | Additional gear ratio – driving shaft revolutions |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 |
| Description | 1st additional gear ratio - driving shaft |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |
| Subindex | 2 |
| Description | reserved |
| Category | |
| Access | |
| PDO mapping | |
| Value range | |
| Default value | |
| Subindex | 3 |
| Description | 3rd additional gear ratio - driving shaft |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |

21.3.95 Object 60EEh: Additional feed constant - driving shaft revolutions

This object provides the driving shaft revolutions for the additional feed constant calculation. This object shall be used with the corresponding subindex of the object 60E9h (feed for the additional feed constant calculation). The value of the object 60E4h is calculated analog to the feed constant calculation for the actual position 6064h via object 6092h.

| | |
|---------------|--|
| Index | 60EEh |
| Name | Additional feed constant - driving shaft revolutions |
| Object code | ARRAY |
| Data type | UNSIGNED32 |
| Category | optional |
| Subindex | 0 |
| Description | highest sub-index supported |
| Category | mandatory |
| Access | R/O |
| PDO mapping | not possible |
| Value range | 3 |
| Default value | 3 |
| Subindex | 1 |
| Description | 1st additional feed constant – driving shaft revolutions |
| Category | mandatory |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |
| Subindex | 2 |
| Description | reserved |
| Category | |
| Access | |
| PDO mapping | |
| Value range | |
| Default value | |
| Subindex | 3 |
| Description | 3rd additional feed constant – driving shaft revolutions |
| Category | optional |
| Access | R/W |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 1 |

21.3.96 Object 60F4h: Following error actual value (DS402)

This object returns the current value of the following error in units defined by the user.

| | |
|---------------|------------------------------|
| Index | 60F4h |
| Name | Following error actual value |
| Object code | VAR |
| Data type | Integer32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Default value | 0 |

21.3.97 Object 60FCh: Position demand internal value (DS402)

This object provides the output of the trajectory generator in position modes. The value is consistent in scaling to the actual internal position value (6063h) and the first setpoint in object 60C1h.

| | |
|---------------|--------------------------------|
| Index | 60FCh |
| Name | Position demand internal value |
| Object code | VAR |
| Data type | INTEGER32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | INTEGER32 |
| Default value | 0 |

21.3.98 Object 60FDh: Digital inputs (DS402)

This index defines simple digital inputs for drives. The manufacturer bits 16 to 2216 to 18 display the actual state of the digital inputs 1 to 71 to 3 (DINx.STATE). The manufacturer bits 24 to 3024 to 26 latch a state change of the digital inputs 1 to 71 to 3. Bits 24 to 3024 to 26 can be reset with object "20B8h" (→ p. 1040).

| | |
|---------------|----------------|
| Index | 60FDh |
| Name | digital inputs |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | optional |
| Access | R/O |
| PDO mapping | possible |
| Value range | UNSIGNED32 |
| Default value | 0 |

| | | | | | | | |
|-----------------------|----|----------|---|--------|-------------|-------------------|-------------------|
| 31 | 16 | 15 | 4 | 3 | 2 | 1 | 0 |
| manufacturer specific | | reserved | | enable | home switch | pos. limit switch | neg. limit switch |
| MSB | | | | | | | LSB |

21.3.99 Object 60FEh: Digital outputs (DS402)

This index defines simple digital outputs for drives. The manufacturer bits 16 and 17 are bit 16 is show the actual status of the digital outputs 1 and 2 output 1.

| | | | | | | |
|-----------------------|-----------------------------|-------|-------|----------|---|-----------|
| Index | 60FEh | | | | | |
| Name | digital outputs | | | | | |
| Object code | Array | | | | | |
| Data type | UNSIGNED32 | | | | | |
| Category | optional | | | | | |
| Subindex | 0 | | | | | |
| Description | highest sub-index supported | | | | | |
| Category | mandatory | | | | | |
| Access | R/O | | | | | |
| PDO mapping | not possible | | | | | |
| Value range | 2 | | | | | |
| Default value | 2 | | | | | |
| Subindex | 1 | | | | | |
| Description | physical outputs | | | | | |
| Category | mandatory | | | | | |
| Access | R/W | | | | | |
| PDO mapping | possible | | | | | |
| Value range | UNSIGNED32 | | | | | |
| Default value | 0 | | | | | |
| Subindex | 2 | | | | | |
| Description | bit mask | | | | | |
| Category | optional | | | | | |
| Access | R/W | | | | | |
| PDO mapping | not possible | | | | | |
| Value range | UNSIGNED32 | | | | | |
| Default value | 0 | | | | | |
| 31 | 18 | 17 | 16 | 15 | 1 | 0 |
| manufacturer specific | | DOUT2 | DOUT1 | reserved | | set brake |
| MSB | | | | | | LSB |

21.3.100 Object 60FFh: Target velocity (DS402)

The speed setpoint (target velocity) represents the setpoint for the ramp generator.

| | |
|---------------|-----------------------------|
| Index | 60FFh |
| Name | target velocity, VL.CMDU |
| Object code | VAR |
| Data type | INTEGER32 |
| Mode | pv |
| Access | R/W |
| PDO mapping | possible |
| Unit | increments |
| Value range | (-2^{31}) to $(2^{31}-1)$ |
| Default value | — |
| Float scaling | 1000:1 |
| EEPROM | no |

21.3.101 Object 6502h: Supported drive modes (DS402)

A drive can support more than one and several distinct modes of operation. This object gives an overview of the implemented operating modes in the device. This object is read only.

| | |
|---------------|------------------------|
| Index | 6502h |
| Name | supported drive modes |
| Object code | VAR |
| Data type | UNSIGNED32 |
| Category | optional |
| Access | R/O |
| PDO mapping | not possible |
| Value range | UNSIGNED32 |
| Default value | 0xE5 (csp ip hm pv pp) |

| | | | | | | | | | | | | | | |
|-----------------------|----------|-------|-----|-----|-----|----|----|----------|----|----|----|----|---|-----|
| 31 | 16 | 15 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| manufacturer specific | reserved | cstca | cst | csv | csp | ip | hm | reserved | tq | pv | vl | pp | | |
| MSB | | | | | | | | | | | | | | LSB |

22 Glossary

| Term | Description |
|---------------------------|---|
| 60° & 120° Hall Sequences | A method for decoding logic signals produced by Hall effect sensors installed in an electronically-switched three-phase brushless motor. |
| Absolute Encoder | The optical disk of the absolute encoder is designed to produce a digital word that distinguishes N distinct positions of the shaft. For example, if there are 8 tracks, the encoder is capable of producing 256 distinct positions or an angular resolution of 1.406 (360/256) degrees. |
| Absolute Move | A positioning movement which is referenced to a fixed original position. For example, if a stage is positioned at +500 mm, an absolute move to +300 mm would result in a move of 200 mm towards the origin (in the negative direction). |
| Acceleration | A change in velocity over time. Because velocity is a vector , it can change in two ways: a change in magnitude and/or a change in direction. In one dimension, acceleration is the rate at which something speeds up or slows down. However, more generally, acceleration is a vector quantity expressing the change with time of the velocity both in magnitude and in direction. |
| Accuracy | Accuracy is the distance between the actual position of a mechanical system and the expected position. It is typically specified in microns or arcsec per given travel for a deviation of $\pm 3\sigma$ (sigma). |
| Actuator | Mechanical device for moving or controlling a mechanism or system. The actuator is usually a physical mechanism but it can also refer to an artificial agent/intelligent agent. In virtual instrumentation actuators and sensors are the hardware complements of virtual instruments. Computer programs of virtual instruments use actuators to act upon real world objects. |
| Almirant | Almirant is a read only type of screen offered to the end user to control Sinop. |
| AS Input | A safety feature in the drive to prevent servos from being restarted. |
| ASCII | Short for American Standard Code for Information Interchange. ASCII provides a one-to-one mapping between alphanumeric characters, and a digital one-byte word. |
| Astrolab | Astrolab is a Windows integrated design environment (IDE) containing all the tools and editors (based on the different IEC 61131 languages) that users need during the entire life cycle of the machine. |
| Auxiliary Encoder | Describes the feedback on the load to close the position loop in critical applications. See Encoder. |
| Backlash | The backlash is an error in positioning caused by the reversal of travel direction. It is caused by a clearance between the elements of the mechanical system. The backlash also affects the bidirectional repeatability. The backlash can be compensated by the position controller. |
| Bandwidth | In computer networking, bandwidth often refers to a data rate measured in bits/s, for example, network throughput. The reason for the connection of data rate with the term <i>bandwidth</i> is that the limit to the data rate of a physical communication link is related to its bandwidth in hertz. |
| Bandwidth | The frequency at which the small signal response of a servo system begins to drop off; stage performance is enhanced by increasing servo bandwidth, although structural resonances typically limit the achievable bandwidth. |

| Term | Description |
|---------------------------|---|
| Blackfin | Blackfin refers to a family of 16/32-bit microprocessors with built-in Digital Signal Processor (DSP) functionality, which is traditionally only accompanied by a small and power-efficient microcontroller. The result is a low-power, unified processor architecture that can run operating systems while simultaneously handling complex numeric tasks such as real time H.264 video encoding. |
| Brake Applied or Released | The two states of the braking system. A device for slowing or stopping the motion of a motor, or alternatively a device to restrain it from starting to move again. |
| Brushless servomotor | An "inside-out" DC motor, with a permanent magnet rotor, and electrical coils in the stator. Commutation of current in the windings is typically achieved via external switching transistors, and Hall-effect detectors. This avoids the limited life of brushes and their radiated EMI. |
| Busy/Free | AKD PDMM support only one connection at a time. If a connection is in use, then the drive is "busy." If the connection is not use, then the drive is "free." |
| CAM profiling | The position of a slave axis is mathematically linked to the position of a master axis. A good example of this would be in a system where two rotating drums turn at a given ratio to each other. A more advanced case of electronic gearing is electronic camming. With electronic camming, a slave axis follows a profile that is a function of the master position. This profile need not be linear, but it must be a mathematical function. |
| CANopen | CANopen is a communication protocol and device profile specification for embedded systems used in automation for fieldbuses working in real-time |
| Cantilevered load | Any load not symmetrically mounted on a stage. Such loads exert torque moments upon the ways, and the resulting deformations can degrade accuracy |
| Caret | The term caret is also sometimes used in graphical user interface terminology where it means a text insertion point indicator, frequently represented by a blinking vertical bar. In this context, it may be used interchangeably with the word cursor , although the latter term is often reserved for a mouse pointer |
| Casting | see Typecasting item |
| Closed loop positioning | The use of feedback devices (encoders, resolvers, interferometers, etc.) to allow a motor to position a user payload accurately. |
| CmdGain | Sets the command gain for the command input. |
| CmdOffset | Sets the command offset for the command input. |
| CmdSrc | Selects Analog, PWM, or Command variable for command. |
| Coefficient of friction | This is defined as the ratio of the force required to move a given load to the magnitude of that load. Typical values for ball and crossed roller slides are 0.001 to 0.005. |
| Cogging compensation | Cogging compensation can be used to counteract cogging effects present in the motor by supplying an additional feedforward current. The result is a reduction of velocity ripple, mainly at lower velocities. |
| COM | COM is the original name of the serial port interface. It might not only refer to physical ports, but also to virtual ports, such as ports created by Bluetooth or USB-to-Serial adapters. |
| Contactors | A contactor is an electrically controlled switch (relay) used for switching a power circuit. A contactor is activated by a control input which is a lower voltage / current than that which the contactor is switching. Unlike a circuit breaker a contactor is not intended to interrupt a short circuit current. |

| Term | Description |
|----------------|--|
| Custom | Refers to user modified or defined material (such as custom units or custom settings) |
| DeviceNet | DeviceNet is a communication protocol (based on Controller Area Network) used in the automation industry to interconnect control devices for data exchange. Typical applications as listed: information exchange, safety devices, and large I/O control networks |
| Drive | In electrical engineering, a drive is an electronic device to provide power to a motor or servo. |
| Driver | In computing and electronics, a driver is a software component allowing higher-level computer programs to interact with a computer hardware device. A driver typically communicates with the device through the computer bus or communications subsystem to which the hardware is connected. |
| Eccentricity | The eccentricity is the distance between the centring diameter of the baseplate and the centring diameter of the rotor. It is not dependent on the position. |
| Encoders | Generates an output signal that is directly proportional to the movement of the motor shaft. This signal is fed into the control circuitry in order to control the shaft position and speed. See also Optical encoders. The most common types are: <ul style="list-style-type: none"> • Incremental/Serial encoders • Hall effect sensors • Resolvers • Tachometer Generators |
| Encoder Counts | Used after quadrature decode. |
| Encoder Lines | Used before quadrature decode. |
| Environment | Environment objects are global objects that exist before the execution of the script. Typically, they are global objects of Astrolab that can be accessed from the script. |
| EtherCAT | "Ethernet for Control Automation Technology" EtherCAT is an open high performance Ethernet-based fieldbus system. The development goal of EtherCAT was to apply Ethernet to automation applications which require short data update times (also called cycle times) with low communication jitter (for synchronization purposes) and low hardware costs. |
| Ethernet | Ethernet is a large, diverse family of frame-based computer networking technologies that operate at many speeds for local area networks (LANs). |
| EtherNet/IP | An open industrial application layer protocol for industrial automation applications. The EtherNet/IP application layer protocol is based on the CIP layer. |
| Fast Input | The inputs are taken into account at each cycle depending on the system periodicity (for instance each Millisecond). Under certain circumstances this can be insufficient when we need more accuracy or if a quick respond from the system is required. To fill in the gap, a drive may have some fast input connections (generally one or two). When an event happens that triggers a Fast Input (e.g. when a sensor sends a rising edge), the detection of a signal occurs faster (which can be 1000-times more accurate than the system periodicity). Then the timestamp associated to this input can be provided to the IPC for taking the corrective actions. |

| Term | Description |
|---|---|
| Feedback Device | Some systems use a feedback device connected to the motor shaft or part of the driven mechanism to control the direction, acceleration and speed of the motor. |
| Fieldbus | A fieldbus is an industrial network system for real-time distributed control (e.g. CAN, Profibus, sercos®...). It is one of the ways to connect Instruments in a plant design. |
| Flatness of travel | Deviation from ideal straight line travel in a vertical plane, also referred to as vertical runout. |
| Frame | In networking dialect, a message is called a frame. |
| Front-end | In software design, the front-end is the part of a software system that interacts directly with the user. |
| FW | Firmware. |
| G.p.i.b. | A standardized protocol, analogous to RS-232, for transmitting digital information. Unlike RS-232, the GPIB interface (also called IEEE-488) transmits data in parallel, not serial format (one byte at a time). |
| Half step | An extended 8-cycle current switching sequence for stepping motors that doubles resolution, reduces noise, and improves resonant conditions. |
| Hall-effect sensor | Highly accurate, non-contact limit switch which detects the proximity of a magnet and provides a digital output to assure an accurate position reference. Typical repeatability of the NEAT Hall sensor is $\pm 1-2$ microns. |
| Holding torque | Stepping motors, when energized, hold position via a magnetic field. The holding torque is the maximum torque which can be generated before the rotor slips to the next pole location (7.2 degrees for 200 step/revolution steppers). |
| Home switch | Any of a variety of sensors which can be used to establish an accurate initial position. This may consist of a standard end-of-travel Hall sensor; a center position opto-interrupter with half-travel blocking vane; an index signal on a linear encoder; a shaft coupling mounted magnet with Hall sensor; or a once-per-revolution encoder index signal. Once-per-revolution sensors will usually require a logical or-ing with a linear signal if a unique home position is required. |
| Homing | The homing procedure allows, based on a position measurement, to set a position offset to the motor in order to ensure it is physically at the home position. |
| Horizontal Straightness | The horizontal straightness is a horizontal deviation from the straight line of travel. A horizontal straightness deviation in the travel of the X-axis will cause a positioning error in the Y direction. |
| Hysteresis Error | The hysteresis error is a deviation between the actual and the commanded position caused by elastic forces accumulated in the motion system. It affects the accuracy and the bidirectional repeatability. |
| HW | Hardware |
| Incremental encoder (Relative encoder) | The incremental encoder, sometimes called a relative encoder, is simpler in design than the absolute encoder. It consists of two tracks and two sensors whose outputs are called channels A and B. As the shaft rotates, pulse trains occur on these channels at a frequency proportional to the shaft speed, and the phase relationship between the signals yields the direction of rotation. |
| Incremental move | Positioning mode in which moves are referenced from the previously held position. |
| Index | Zero pulse (zero signal) of a hardware pin such as digital input or feedback pin. |

| Term | Description |
|------------------|---|
| Index Pulse | A single generated by a second track on the encoder emitter disk that occurs once per revolution, which can be used to indicate an absolute position of the motor shaft. Encoder manufacturers identify the index pulse using different terms such as index, marker, home position and zero reference. |
| Inductance | The tendency of a motor coil to store energy in a magnetic field. High speed stepping motor performance is inversely proportional to motor inductance. |
| Internet socket | An Internet socket (or commonly, a network socket or socket), is an end-point of a bidirectional process-to-process communication flow across an IP based network, such as the Internet. Each socket is mapped to an application process or thread. A socket is an interface between an application process or thread and the TCP/IP protocol stack provided by the operating system. The plug type is generally RJ-45. |
| Interrupt | An interrupt is an asynchronous signal from hardware indicating the need for attention or a synchronous event in software indicating the need for a change to another execution stream. |
| Intime | INtime software combines deterministic, hard real-time control with standard Windows operating systems (including Windows XP, Windows XP Embedded, Windows 2000 and Windows Server 2003) without requiring additional hardware. Intime was designed specifically to take advantage of the powerful capabilities of the x86 processor architecture. Therefore, you're real-time and non real-time applications run in separate virtual machines on a single computer, for cost-effective, reliable control that is easy to develop and maintain. |
| Jerk | In physics, jerk, is the rate of change of acceleration; more precisely, the derivative of acceleration with respect to time |
| Lead error | The deviation of a leadscrew from its nominal pitch. The error is often monotonic (linear), although periodic error and thermal expansion set limits to its predictability. |
| Limit switch | A sensor, typically Hall-effect, optical, eddy current, or mechanical, which is used to sense the end of travel of a linear motion assembly. In addition to preventing overtravel, it is frequently used to establish a precision reference. |
| Mechanical angle | Mechanical angle of the feedback (always between 0°...360°) |
| Mechanical angle | Mechanical angle of the feedback, always between 0° and 360°. |
| Microstepping | A technique which, instead of switching phase currents in a stepping motor on and off, sinusoidally varies the current in the two windings. This effectively increases the resolution from 200 steps per revolution to 2,000 (~10) or 10,000 (~50) micro-steps per revolution. |
| Modbus | Modbus is a serial communications protocol and is now the most commonly available means of connecting industrial electronic devices. Modbus is often used to connect a supervisory computer with a remote terminal unit in supervisory control and data acquisition (SCADA) systems. Versions of the Modbus protocol exist for serial port and Ethernet (it is widely used with TCP/IP over Ethernet) |

| Term | Description |
|--------------------------------|---|
| Motion control | Motion control is a sub-field of automation, in which the position and/or velocity of machines are controlled using some type of device such as a hydraulic pump, linear actuator, or an electric motor, generally a servo. Motion control is an important part of robotics and CNC machine tools, however it is more complex than in the use of specialized machines, where the kinematics are usually simpler. The latter is often called General Motion Control (GMC). Motion control is widely used in the packaging, printing, textile and assembly industries |
| Motor Continuous Stall Current | The amount of current applied to a motor when the rotor is locked and results in the rated temperature rise. |
| Motor Peak Current | Peak current or amps peak is the highest current that can flow through the motor or $\text{amps peak} = 1.41 \times \text{amps/phase (or amps RMS)}$. |
| Nonvolatile | Information is stored in a specific memory to remain accessible even when the application has been powered off. The memory type is called NV (Non-Volatile Memory) also NVS (Non Volatile Storage). |
| Nut stiffness | The stiffness of a leadscrew/nut assembly, typically measured in Newtons per meter (N/m). This stiffness, together with the moving mass and duplex bearing stiffness, sets the primary natural frequency of a leadscrew-driven stage. |
| Online/Offline | Active or connected/inactive or disconnected. |
| Open loop positioning | A positioning technique, typically utilizing stepping motors, in which the controller issues a sequence of commands to the motor without any absolute means of detecting if the move has in fact been made. When the load and move velocity and acceleration are appropriately defined, open loop positioning is capable of extended operation without losing steps. |
| OpenGL | OpenGL (Open Graphics Library) is a standard specification defining a cross-language cross-platform API for writing applications that produce 2D and 3D computer graphics. The Softscope uses this API to implement graphical manipulations. |
| Optical encoder | A linear or angular position feedback device, typically providing incremental two channel information in quadrature format (sine or square waves with a 90 degree phase shift between each channel). Such two channel information allows simple counter circuits to function as absolute position indicators. |
| Orthogonality | The degree of perpendicularity, or squareness, between the two axes in an X-Y or X-Z table. This parameter is usually measured in arc-seconds or micro radians. |
| P-code | a P-code machine or pseudo-code machine is a specification of a CPU (Central Processing Unit) whose instructions are expected to be executed in software rather than in hardware. Programs that have been translated to p-code are executed (interpreted) by a software program that emulates the behavior of the CPU specification. |
| Periodicity | Motor systems having a reciprocating or oscillating motor that operates synchronously with the periodicity of the source which supplies the electrical energy. The period of execution of a pipe is the time spent between two successive computations of set values for the same pipe. The period of execution of a pipe is specified by the PERIOD parameter of the input pipe block. |
| Phase current | The rated current which a stepping motor requires to generate its rated holding torque. This value is usually based on unipolar (half-coil) operation. This choice of how the motor is wired has significant impact on performance. |

| Term | Description |
|-----------------|---|
| Phase sequence | The specific sequence of coil current changes used to advance a stepping motor clockwise and counter-clockwise, in either full or half step modes. |
| Pitch | <p>For lead-screws specified in British units, the number of full rotations required to advance the nut 1f. For example, a 5 pitch lead-screw has a lead of 0.200f. Metric screws are specified by lead only, in millimeters.</p> <p>The pitch is a rotation (angular deviation from the ideal straight line of travel) around an axis in the horizontal plane perpendicular to the direction of travel. For the X-axis travel, a pitch will cause an Abbé error in the X and Z directions</p> <p>An angular deviation possible in positioning systems, in which the table's leading edge rises or falls as the table translates along its direction of travel. This represents rotation around a horizontal axis, perpendicular to the direction of travel.</p> |
| PLCopen | A vendor -and product- independent worldwide association active in Industrial Control and aiming at standardizing PLC file formats based on XML |
| Position counts | AKD PDMMscales this so that the low 32 bits are the position per revolution (mechanical or electrical?). |
| Pragma | A compiler directive communicating additional "pragmatic" information. |
| Precedence | <p>In arithmetic and algebra, when a number or expression is both preceded and followed by a binary operation, a rule is required for which operation should be performed first. From the earliest use of mathematical notation, multiplication took precedence over addition, whichever side of a number it appeared on.</p> <p>Thus $3 + 4 \times 5 = 5 \times 4 + 3 = 23$.</p> <p>To change the order of operations, we use parentheses (). Thus, if we want to force addition to precede multiplication, we write $(3 + 4) \times 5 = 35$.</p> |
| Primary Encoder | Describes the feedback mounted onto the motor the drive is controlling. The Primary Encoder is used to force commutation. See also Encoder. |
| Probe | See Softscope - Probe item |
| PROFIBUS | <p>PROFIBUS (Process Fieldbus) is a standard for fieldbus communication in automation technology. PROFIBUS is the only fieldbus that can be used in equal measure in production automation and process automation and has since become a global market leader. Worldwide</p> <p>There are two variations of PROFIBUS; the most commonly used DP, and the lesser used PA variations:</p> <ul style="list-style-type: none"> • PROFIBUS DP (Decentralized Peripherals) is used to operate sensors and actuators via a centralized controller in production technology. The many standard diagnostic options, in particular, are focused on here. Other areas of use include the connection of "distributed intelligence", i.e. the networking of multiple controllers to one another (similar to PROFIBUS FMS). Data rates up to 12 Mbit/s on twisted pair cables and/or fiber optics are possible. • PROFIBUS PA (Process Automation) is used to monitor measuring equipment via a process control system in process engineering. This PROFIBUS variant is ideal for explosion-hazardous areas (Ex-zone 0 and 1). Here, a weak current flows through bus lines in an intrinsically safe circuit so that explosive sparks are not created, even if a malfunction occurs. The disadvantage of this variant is the slower data transmission rate of 31.25 kbit/s. |
| Programming | The act of performing a sequence of instructions or commands. |

| Term | Description |
|--------------------------------|---|
| Quasi-closed loop positioning | A technique using a stepping motor and encoder in which open loop moves are completed, after which an encoder/counter is checked and, if necessary, a small final move or moves is used to achieve the desired accuracy. |
| Radial Run-out | The radial runout is the positioning error of a rotary table's centering diameter in the horizontal direction when the table is rotating in the horizontal plane. |
| Ramping | The gradual acceleration and deceleration of a stepping motor, essential if performance beyond the start/stop range is required. The slope of the ramp is a function of screw pitch, load, drive voltage and design, and motor. |
| Real-time computing | RTC is the study of hardware and software systems which are subject to a "real-time constraint" (i.e. operational deadlines from event to system response). By contrast, a non-real-time system is one for which there is no deadline, even if fast response or high performance is desired or even preferred. The needs of real-time software are often addressed in the context of real-time operating systems, and synchronous programming languages, which provide frameworks on which to build real-time application software. A real time system may be one where its application can be considered (within context) to be mission critical. |
| Reference Counting | In computer science, reference counting is a technique of storing the number of references, pointers, or handles to a resource such as an object or block of memory. It is typically used as a means of de-allocating objects which are no longer referenced. |
| Reference situation | Any situation that indicates that the home position has been found. |
| Reference switch | An adjustable hardware switch for indicating a home position. |
| Refresh | GUI command that re-loads the data from the drive and redraws the display. |
| Regen resistor | Regen Resistors are a necessity on all horizontal motion (bridge/trolley/monorail) applied Variable Speed drives. Power returns to the drive when the connected motor is driven in generator mode and is called regenerative power. This power is absorbed by charging the bus smoothing capacitor with the regen resistor consuming all that regenerative power exceeding the chargeable energy of the capacitor. |
| Relative encoder | See Incremental encoder. |
| Repeatability, bi-directional | The difference in absolute position reached when returning to a given position from the same direction. This value may mask significant amounts of backlash. |
| Repeatability, uni-directional | The difference in absolute position reached when returning to a given position from the opposing direction. This value is usually larger than the unidirectional repeatability. |
| Resolution | The resolution is the smallest possible movement that can be achieved by a system. It can be defined at the electronics, encoder and mechanics level. The distance a stage can be commanded to move in a single step. For servo systems, the basic increment produced by its optical encoder, or any other feedback device. |
| Resonance, midrange | A parasitic oscillation which is endemic to stepping motors, although frictional loads may mask its effect. It typically sets in from 5-15 revolutions per second, and can easily cause a loss of synchronization (stalling). All NEAT high speed micro-stepping drives effectively suppress this resonance. |

| Term | Description |
|-----------------------|--|
| Resonance, primary | The rotor inertia of a stepping motor, together with its spring-like holding torque, constitutes a basic spring-mass oscillator. In the absence of sufficient damping, stepping at certain frequencies may excite resonance in this system, or resonate with the stage or load, resulting in loss of synchrony. The addition of system damping, micro-stepping, or ramping through problematic speeds will usually eliminate this resonance. |
| Rising Edge | A rising edge is the transition of a digital signal from low to high. It is also named positive edge. |
| RJ-45 | Internet plug type. |
| Roll | An angular deviation from an ideal straight line motion, in which the positioning table rotates around its axis of travel as it translates along that axis. |
| RS-232c | A popular protocol for transmitting digital data over two lines in a bit-serial format. RS-232C specifies signal levels, data formats, maximum transmission distance, etc. |
| Runtime | In computer science, runtime or run time describes the operation of a computer program, the duration of its execution, from beginning to termination (compare compile time). |
| Safe Torque Off | A safety feature in the drive to prevent servos from being restarted (STO). |
| Sensor | A sensor is a type of transducer that converts one type of energy to another for various purposes including measurement or information transfer. |
| sercos® | |
| Servo Drive | A servo drive is a special electric amplifier used to power electric servo motors. It monitors feedback signals from the motor and continually adjusts for deviation from expected behavior. |
| Servo Motor | A DC motor which produces a torque proportional to current. Precise positioning is achieved by linear or PWM (duty cycle) control of motor current or voltage, together with accurate monitoring of position via an external feedback device. |
| Setpoint | Setpoint is the target value that an automatic control system, for example PID controller, will aim to reach |
| Sinop | Sinop is a runtime engine issued from Astrolab and that provides a soft PLC and a motion controller via a graphical view of Axis movement with log messages |
| Softscope - Channel | A Channel is used by the softscope to acquire the evolution of a variable that is plugged into it |
| Softscope - Probe | A device that uses onboard instruments to gather and relay a variety of measurement to controllers from remote locations. Probes may return their data over radio links or be physically tethered to controllers or another device, or to collect and return physical samples |
| Softscope - Sampling | To acquire the variable's evolution, samples are taken at fixed intervals. The accuracy to create the trace depends on the resolution of the acquisition. The sampling frequency must be higher than 2 times the highest frequency in the input signal. This is called the Nyquist frequency. Theoretically it is possible to reconstruct the input signal with more than 2 samples per period. In practice, 10 to 20 samples per period are recommended to be able to examine the signal thoroughly |
| Softscope - Time base | The time base allows to set the speed at which all the lines for each channel are drawn, and is calibrated in milliseconds per division |

| Term | Description |
|------------------------|--|
| Softscope - Trace | The trace is the resulting graph of a variable's evolution against time, with the more distant past on the left and the more recent past on the right |
| Stall speed | The maximum speed which a stepping motor, properly ramped, can achieve without loss of synchrony. This speed is a function of motor inductance, ramp slope, applied load, and drive voltage and design. |
| Start/stop speed | The maximum step rate which can be applied to a stationary stepping motor and still retain error-free performance. Also, the rate from which a stepping motor may be instantaneously stopped without overshooting. This is a function of the screw pitch, load, drive voltage and design, and motor. |
| Step rate | A type of motor featuring two or four stator coils and a toothed permanent magnet rotor, which moves through a small angle in response to a specific sequence of coil current changes. |
| Stepping motor | The frequency of coil current changes, or input pulse train, applied to a stepping motor, in pulses/second or hertz. For 200 step/revolution motors, the full step rate multiplied by 0.3 equals the rotation rate in R.P.M. |
| Straightness of travel | Deviation from straight line motion in a horizontal plane. Also referred to as horizontal runout. This error is usually traceable to an underlying angular error of the ways. |
| SynqNet | SynqNet is a digital machine control network. Built on the 100BT physical layer, SynqNet provides a synchronous real-time connection between motion controllers, servo drives, stepper drives, I/O modules, and custom devices |
| T.i.r. | This stands for Total Indicator Reading, which reflects the total absolute deviation from a mean value (versus a \pm value which indicates the deviation from a nominal value). |
| Thermal Expansion | The thermal expansion is a change of the size and shape of a system when the temperature is modified. The amount of change is dependent on the size of the component, the degree of temperature change and the characteristics of the material. |
| Timestamp | A timestamp is a sequence of characters, denoting the date and/or time at which a certain event occurred. |
| Torque | A radially directed force, typically measured in inch-ounces, foot-lbs. or Newton-meters. |
| Typecasting | In computer science, type conversion or typecasting refers to changing an entity of one data type into another. This is done to take advantage of certain features of type hierarchies. For instance, values from a more limited set, such as integers, can be stored in a more compact format and later converted to a different format enabling operations not previously possible, such as division with several decimal places' worth of accuracy. There are two types of conversion: implicit and explicit. The term for implicit type conversion is coercion. The most common form of explicit type conversion is known as casting. Explicit type conversion can also be achieved with separately defined conversion routines such as an overloaded object constructor. |
| Velocity | A vector quantity whose magnitude is a body's speed and whose direction is the body's direction of motion. Both speed and direction are required to define it and measured in meters/second (m/s). |
| Wibu | Developed by WIBU-SYSTEMS AG, WibuKey is the way to protect DMS software |

| Term | Description |
|------------|---|
| Wobble | To move or rotate with an uneven or rocking motion or unsteadily from side to side. |
| Yaw | The yaw is a rotation (angular deviation from the ideal straight line of travel) around the Z axis which is a vertical plane perpendicular to the direction of travel. For the X-axis travel, the yaw will cause an Abbé error in the X and Y directions. An angular deviation from ideal straight line motion, in which the positioning table rotates around the Z (vertical) axis as it translates along its travel axis. |
| Zero angle | Mechanical zero angle of the feedback = 0 degree. |

This page intentionally left blank.

23 Index

1

| | |
|------------|-----------|
| 1000h | 844, 1000 |
| 1001h | 845, 1001 |
| 1002h | 846, 1002 |
| 1003h | 847, 1003 |
| 1005h | 848, 1004 |
| 1006h | 848, 1004 |
| 1008h | 849, 1005 |
| 1009h | 849, 1005 |
| 100Ah | 849, 1005 |
| 100Ch | 850, 1006 |
| 100Dh | 850, 1006 |
| 1010h | 851, 1007 |
| 1011h | 852, 1008 |
| 1012h | 853, 1009 |
| 1014h | 853, 1009 |
| 1016h | 854, 1010 |
| 1017h | 855, 1011 |
| 1018h | 855, 1011 |
| 1026h | 857, 1013 |
| 1400-1403h | 858, 1014 |
| 1600-1603h | 859, 1015 |
| 1800-1803h | 861, 1017 |
| 1A00-1A03h | 864, 1020 |
| 1C12h | 865, 1021 |
| 1C13h | 866, 1022 |

2

| | |
|------------|-----------|
| 2000h | 867, 1023 |
| 2001h | 867, 1023 |
| 2002h | 868, 1024 |
| 2011h | 869, 1025 |
| 2012h | 870, 1026 |
| 2013h | 871, 1027 |
| 2014-2017h | 872, 1028 |
| 2018h | 873, 1029 |
| 2026h | 875, 1031 |
| 204Ch | 876, 1032 |
| 2071h | 877, 1033 |
| 2077h | 877, 1033 |
| 20A0h | 877, 1033 |
| 20A1h | 879, 1035 |
| 20A2h | 879, 1035 |
| 20A3h | 880, 1036 |
| 20A4h | 880, 1036 |
| 20A5h | 881, 1038 |
| 20A6h | 881, 1038 |
| 20A7h | 883, 1040 |
| 20B8h | 883, 1040 |

3

| | |
|-------|-----------|
| 345Ah | 884, 1041 |
| 3474h | 887, 1044 |

| | |
|-------------|-----------|
| 3475h | 888, 1045 |
| 3496h | 889, 1046 |

6

| | |
|-------------|-----------|
| 6040h | 891, 1048 |
| 6041h | 892, 1049 |
| 605Ah | 895, 1052 |
| 6060h | 896, 1053 |
| 6061h | 897, 1054 |
| 6063h | 897, 1054 |
| 6064h | 897, 1054 |
| 6065h | 898, 1055 |
| 606Ch | 898, 1055 |
| 6071h | 899, 1056 |
| 6073h | 899, 1056 |
| 6077h | 899, 1056 |
| 607Ah | 900, 1057 |
| 607Ch | 900, 1057 |
| 607Dh | 901, 1058 |
| 6081h | 902, 1059 |
| 6083h | 902, 1059 |
| 6084h | 902, 1059 |
| 608Fh | 904, 1061 |
| 6091h | 905, 1062 |
| 6092h | 906, 1064 |
| 6098h | 907, 1065 |
| 6099h | 908, 1066 |
| 609Ah | 910, 1068 |
| 60B1h | 910, 1068 |
| 60B2h | 910, 1068 |
| 60B8h | 911, 1069 |
| 60B9h | 913, 1071 |
| 60BAh | 914, 1072 |
| 60BBh | 914, 1072 |
| 60BCh | 914, 1072 |
| 60BDh | 915, 1073 |
| 60C0h | 915, 1073 |
| 60C1h | 916, 1074 |
| 60C2h | 917, 1075 |
| 60C4h | 918, 1076 |
| 60D0h | 920, 1078 |
| 60E0h | 921, 1079 |
| 60E1h | 921, 1079 |
| 60E4h | 922, 1080 |
| 60E8h | 923, 1081 |
| 60E9h | 924, 1082 |
| 60EDh | 925, 1083 |
| 60EEh | 926, 1084 |
| 60F4h | 927, 1085 |
| 60FCh | 928, 1086 |
| 60FDh | 928, 1086 |
| 60FEh | 929, 1087 |
| 60FFh | 930, 1088 |
| 6502h | 930, 1088 |

A

| | |
|-------------------------|----------|
| Acceleration ramp | 366 |
| AL Event | 780, 934 |

| | |
|------------------------------------|-----------|
| Analog inputs | 226 |
| Analog setpoints | 226 |
| Anti-resonance filter | 726 |
| Aux. supply 24V, interface | 210 |
| B | |
| Basic drive setup | 34 |
| Blink | 369 |
| Brake view | 64 |
| C | |
| CANopen over EtherCAT | 785, 939 |
| Capture | 64 |
| Cogging compensation | 149 |
| Command source | 375 |
| Connection Diagrams | 206 |
| M variants | 207 |
| Control word | 891, 1048 |
| Control Word Commands | 786, 940 |
| Controlled Stop | 80 |
| Current loop gain scheduling | 84 |
| CW/CCW Input | 230 |
| Cycle Time | |
| Adjust | 799, 953 |
| Max. Values | 799, 953 |
| Cyclical Values | 797, 951 |
| D | |
| DC bus link, interface | 216 |
| deceleration | 379 |
| Direct DC mains | 41 |
| Disable | 385 |
| Dynamic Braking | 82 |
| E | |
| EEProm Content | 810, 965 |
| Emergency Messages | 812, 968 |
| Emergency Off | 82 |
| Emergency Stop Function | 82 |
| emergency timeout | 388 |
| Enable/Disable | 77 |
| Error Messages | |
| parameter and command | 199 |
| EtherCAT Profile | 778, 932 |
| F | |
| Fault messages | 177 |
| Faults | 373 |
| clearing | 199 |
| Feedback 1 | 54 |
| Feedback Connection | 231 |
| Fieldbus | 964 |
| Fieldbus Parameters | 964 |
| Foldback | 59 |

I

| | |
|------------------------------|-----------------|
| I/O-Connection | 222 |
| Inputs | |
| Analog | 226 |
| Interrupt Event | 780, 934 |

L

| | |
|--------------------------|-----------------|
| Latch Words | 802, 956 |
| Limits | 74 |

M

| | |
|--------------------------------------|-----------------|
| Macros | 170 |
| Mailbox | 804, 958 |
| Mains supply, interface | 219 |
| Motor | 50 |
| Motor interface | 211 |
| Motor Power Connection | 211 |

N

| | |
|---|-----------|
| Non-Plug and Play Feedback Devices | 58 |
|---|-----------|

O

| | |
|---|------------------|
| Object Dictionary | 818, 974 |
| Objects sorted | 844, 1000 |
| Operating mode | 896, 1053 |
| parameters and commands active in | 252 |
| Operation Modes | 799, 953 |
| Outputs | |
| Analog | 225 |

P

| | |
|---|-----------------|
| Parameter Load/Save | 175 |
| PDO Fixed Mapping | 789, 943 |
| PDO Flexible Mapping | 791, 945 |
| Performance Servo Tuner | 104 |
| advanced | 108 |
| Phase run-up | 782, 936 |
| Power | 41 |
| Pulse Direction, interface | 229 |

R

| | |
|--|------------------|
| Regen resistor, interface | 214 |
| Regeneration | 43 |
| Response monitoring | 850, 1006 |

S

| | |
|-----------------------------|------------|
| scope | 154 |
| channels | 154 |
| settings and presets | 164 |
| time-base | 154 |
| Service Motion | 89 |
| Settings | 37 |

| | |
|------------------------------|------------------|
| Slave Register | 779, 933 |
| Slider Tuning | 103 |
| State Machine | 785, 939 |
| Status word | 892, 1049 |
| Status Word | 788, 942 |
| Stop Function | 82 |
| Synchronization | 801, 955 |

T

| | |
|-------------------------------|------------|
| Terminal | 169 |
| Trouble Shooting | 204 |
| Tuning | 132 |

W

| | |
|-----------------------|------------|
| Warnings | 177 |
| Wizard | 36 |

About KOLLMORGEN

Kollmorgen is a leading provider of motion systems and components for machine builders. Through world-class knowledge in motion, industry-leading quality and deep expertise in linking and integrating standard and custom products, Kollmorgen delivers breakthrough solutions that are unmatched in performance, reliability and ease-of-use, giving machine builders an irrefutable marketplace advantage.



Join the [Kollmorgen Developer Network](#) for product support. Ask the community questions, search the knowledge base for answers, get downloads, and suggest improvements.

North America KOLLMORGEN

203A West Rock Road
Radford, VA 24141
USA

Web: www.kollmorgen.com
Mail: support@kollmorgen.com
Tel.: +1 - 540 - 633 - 3545
Fax: +1 - 540 - 639 - 4162

South America KOLLMORGEN

Avenida Tamboré, 1077 Tamboré -
Barueri - SP Brasil
CEP: 06460-000

Web: www.kollmorgen.com
Tel.: +55 11 4191-4771

Europe KOLLMORGEN Europe GmbH

Pempelfurtstraße 1
40880 Ratingen
Germany

Web: www.kollmorgen.com
Mail: technik@kollmorgen.com
Tel.: +49 - 2102 - 9394 - 0
Fax: +49 - 2102 - 9394 - 3155

China and SEA KOLLMORGEN

Room 202, Building 3, Lane 168,
Lin Hong Road, Changning District
Shanghai

Web: www.kollmorgen.cn
Mail: sales.china@kollmorgen.com
Tel.: +86 - 400 661 2802

KOLLMORGEN®

Because Motion Matters™