AKD™

User Guide

Edition: May 2013, Revision J Working Draft

Valid for Firmware version 1.9

Patents Pending

Part Number 903-200006-00

c UL us LISTED IND. CONT. EQ. 86F2

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™

Record of Document Revisions:

Revision	Remarks
6/2010	Combined User Guide and Parameter and Command Reference Guide (released in WorkBench only).
9/2010	Combined User Guide, CANopen, and EtherCAT manuals. Updated for Release 1.3.
11/2010, Rev A	Usability improvements, minor topic edits.
12/2010, Rev B	Document part number update.
05/2011, Rev C	Updated for release 1.4. Added Modbus, modulo, new homing modes, registration moves, W&S improved, new parameters, PLS enhanced, velocity unit changes, motion tasking revisions, switch bounce conditioning (DINx.FILTER), linear motor parameters.
10/2011, Rev D	Updated for release 1.5. Added Configuring with Linear Motors section. New parameters for EIP, DRV, MODBUS, VM, DIO, IP, FB3, IL, FB2, AIN. Improved description of rotary switch use, modbus scaling, homing.
03/2012, Rev E	Updated for release 1.6. Added chapter on Using AKD in a Vertical Axis. Updated parameters for FB1, FB2, FB3, CANOPEN. Updated Modbus Manual.
05/2012, Rev F	Updated for BASIC IO option card. Rotary switch functions updated. Option IO parameters in SD, AIN2, AOUT2, DIN, and DOUT parameter categories added.
08/2012, Rev G	Updated with new Wake and Shake chapters and parameters, new Z pulse digital output mode, and Motion Profile Table screens.
12/2012, Rev H	Updated with new parameters for induction motor support. Also added new parameters in FBUS, REGEN, and DRV. Added two new digital output modes, 20 and 21.
05/2013, Rev J	Updated with new FB1 and IL parameters, multiturn overflow features. New motor temperature screen and motor derating.

Hardware Revision (HR)

Hardware Revision	Recommended Firmware	Minimum Firmware	Recommended Workbench	Minimum Workbench	Remarks
A	M_01-06	M_01-01-00-001	1.6.0	1.1.0	Initial Revision
C	M_01-06	M_01-03-00-011	1.6.0	1.3.0	STO certified
D	M_01-08	-	1.8.0	-	Control board revision 9

Trademarks

- AKD is a registered trademark of Kollmorgen Corporation
- 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
- 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)

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

Printed in the United States of America

Table of Contents

1 About the AKD User Guide

1.1 About this User Guide

This guide describes the operation and use of the AKD drive. Each section details a specific topic related to the use of the product in basic terms which will help you get the most from the product. Each section includes examples to help guide you in setting up and using the various features available in the drive.

This guide is for users who have installed and tested the drive according to the *AKD and AKD PDMM Installation Manual*. The *AKD Installation Manual* is included on the product disk and contains critical safety information.

1.2 Abbreviations

Abbreviation	Meaning
AGND	Analog ground
CE	Communité Européenne
COM	Serial interface for a personal computer
DCOMx	Communication line for digital inputs (with x=7 or 8)
Disk	Magnetic storage (diskette, hard disk)
EEPROM	Electrically erasable programmable memory
EMC	Electromagnetic compatibility
F-SMA	Fiber optic cable connector according to IEC 60874-2
LED	Light-emitting diode
LSB	Low significant byte (or bit)
MSB	Main significant byte (or bit)
NI	Zero pulse
PC	Personal computer
PE	Protective earth
PLC	Programmable logic control
PLL	Phase locked loop
PLS	Programmable limit switch
PWM	Pulse-width modulation
RAM	Random access memory (volatile memory)
R_{Brake}/R_B	regen resistor (also called a regen resistor)
RBext	External regen resistor
RBint	Internal regen resistor
RCD	Residual current device
RES	Resolver
ROD	Incremental encoder (A quad B)
S1	Continuous operation
STO	Safe torque off
Vac	Volts, alternating current
Vdc	Volts, direct current

2 AKD Models

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

The figure below shows part number identification for drive features.

The customization code includes language version of printed material for European countries:

- D000 for German
- E000 for English
- F000 for French
- I000 for Italian

2.1 CC Drive Models

CC drive models allow you to select between EtherCAT, CANopen, or analog operation. This drive model is identified with a new model number of the form AKD-Pxxxxx-NACC-0000 (the CC is the unique identifier).

The CC drive model is fitted with both the EtherCAT (X5 and X6) and CANopen (X12 and X13) fieldbus connectors and a new software parameter (DRV.TYPE (pg 559)) allows you to select which features the drive supports; you cannot use EtherCAT and CANopen at the same time.

3 Initial Drive Setup

3.1 Initial Drive Setup

The AKD 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.

Software Installation and Drive Communication Setup:

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

WorkBench System Requirements

Required Components: Microsoft .NET Framework 2.0

Supported Operating Systems:

- Windows XP
- Windows Vista
- Windows 7

3.2 Display Codes

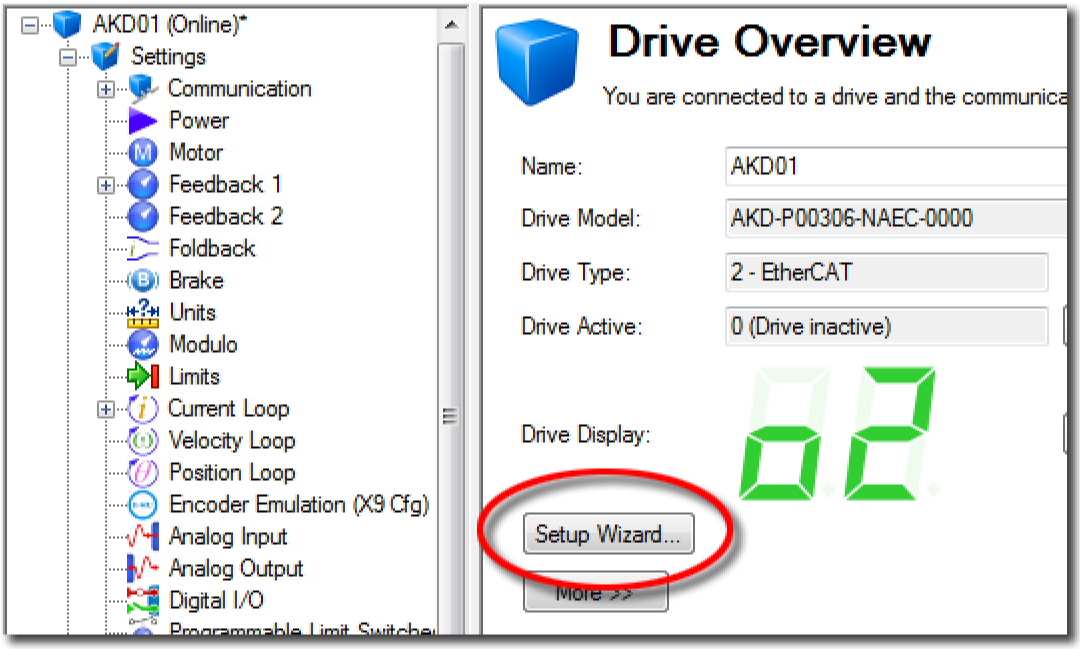
During drive operation the drive display shows the following codes depending on the drive status.

Display Code	Status
o0	Normal operation, current mode, no faults
o1	Normal operation, velocity mode, no faults
o2	Normal operation, position mode, no faults
F [3 digit code, flashing]	Fault (see Fault and Warning Messages)
n [3 digit code, flashing]	Warning (see Fault and Warning Messages)
I,P [IP address]	Displaying drive IP address
- -	Powered on and FPGA loading. If continuous then operational and resident FPGA images are corrupt.
[.]	Drive enabled
[.] (flashing)	Drive in an internal dynamic brake mode (DRV.ACTIVE (pg 491) = 3).
dd	Operational started reboot, waiting for the resifnet to start.
d2	Firmware download: corrupted operational FPGA; resident FPGA is functional.
d3	Firmware download: HW download (HW switch was pressed - Rev 3 and higher).
d4	Firmware download: Corrupted operational FW.
d5	Firmware download: SW download (download command was issued from the operational FW).
d9	Start firmware download
dL	Loading image process is running.

Display Code	Status
dF (flashing)	Failure during firmware download.
Sb	Special mode: Burn-in
Fr	Blackfin Reset
] [	Resident completed and waiting for the operational to start
FP	Kernel Panic
FE	Blackfin Exception

3.3 AKD 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.



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.

4 Connecting the Drive

4.1 Connected and Disconnected States

WorkBench always starts disconnected from any drives. The **Disconnected** view opens when you start WorkBench and offers two choices:

- **Connect**: Opens the **Connect to a Drive** view.
- **Delete**: Opens a list of available drives and allows you to delete a drive from WorkBench.

While WorkBench is trying to establish communications with the drive, WorkBench is in the connecting state. Normally, WorkBench will be in the connecting state for a few moments before the connection is established. If WorkBench cannot establish communications correctly, then a five second timeout occurs and WorkBench returns to the disconnected state.

4.2 Disconnected

When WorkBench is disconnected from a drive, no communication exists between your PC and the drive.

The drive becomes disconnected because of one of the following conditions:

- When WorkBench starts it remembers which drives you were using previously but it does not initially connect to these drives.
- If WorkBench detects that it can no longer communicate with the drive, it will automatically go to this disconnected state. Common causes include a network cable being disconnected or the drive being turned off.
- You pressed the disconnect command.

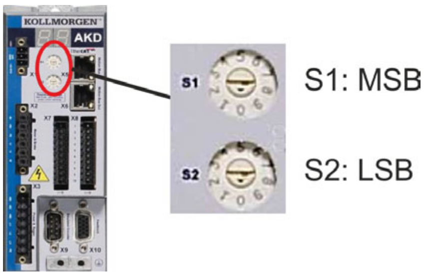
To restore communication:

1. Clicking Connect will start communication with the drive. If WorkBench cannot find the drive, it will immediately return to the disconnected state.
2. Pressing select will show a window where you can select a different drive you would like to use.
3. Pressing delete will remove this drive from the navigation tree on the left hand side of the main window.

4.3 Setting the IP Address AKD-B, AKD-P, AKD-T

The IP address can be flashed across the LED display if the B1 button is pressed.

You can use the rotary switches to set the IP address of the AKD. For CANopen and some other fieldbuses, the rotary switches also set the node address of the drive for that specific network.



Rotary Switch Setting	Drive IP Address
00	DHCP/AutoIP address. The IP address of the drive is obtained from the DHCP server on your network. If no DHCP server is found the IP addresses is an AutoIP address (it is internally generated following the AutoIP protocol and will be of the form 169.254.xx.xx).
01 to 99	Static IP Address. The IP address is 192.168.0.nn, where nn is the number from the rotary switch. This setting generates addresses in a range from 192.168.0.**1**to 192.168.0.**99**. Example:if S1 is set to 2 and S2 is set to 5 – the IP address is 192.168.0.25

NOTE The PC subnet mask must be set to 255.255.255.0 or 255.255.255.128
When connecting the AKD directly to a PC, use static IP addressing (not 00).

Static IP addressing

When connecting the drive directly to a PC, static IP addressing must be used. Set rotary switches S1 and S2 to a number different from 00.
This setting generates addresses in a range from 192.168.0.00**1** to 192.168.0.0**99**.

Dynamic IP addressing (DHCP and Auto-IP)

With S1 and S2 both set to 0, the drive is in DHCP mode. The drive will acquire its IP address from an external DHCP server if present in the network. If a DHCP server is not present, the drive will assume an Automatic Private IP Address of the form 169.254.x.x.

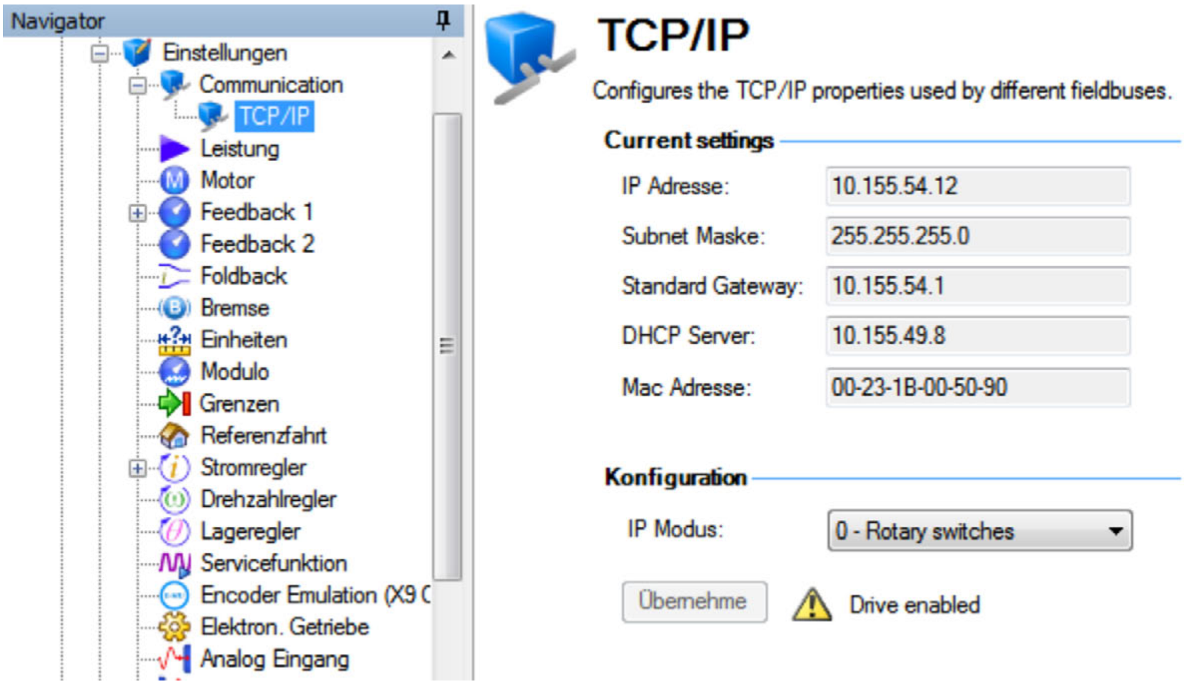
If your PC is directly connected to the drive, and set to obtain an IP address automatically in the TCP/IP settings, a connection will be established with both devices using compatible automatically generated addresses. It can take up to 60 seconds for a PC to configure an Automatic Private IP Address (169.254.x.x).

Changing the IP address

If the switches are altered while 24 V Logic power is supplied to the drive, you must switch off and then switch on again the 24 V supply voltage. This action will reset the address.

4.3.1 Setting IP address with Software

In WorkBench, under Settings > Communication > TCP/IP, the configuration of the IP address can be changed for greater network and fieldbus flexibility. By default, the rotary switch method described above is recommended for simplicity.



There are three modes under IP Mode on the TCP/IP screen by which the IP address can be set.

Mode 0

Rotary Switches (default)

Mode 1

Fixed IP address (insert fixed TCP/IP). Use this mode to set a fixed IP address for the drive which is independent of the rotary switches. This is common with Modbus TCP or Ethernet/IP applications.

Mode 2

DHCP/IP independent of the rotary switches. This is the same behavior as switch setting "00" in Mode 0, however it allows the user to still use the rotary switch settings. For example, CANopen node address is dependant on these switch settings, but the user can now use DHCP/AUtoIP for the IP address setting.

4.3.2 Recovering Communications with a Drive on an Un-Reachable IP Address

Sometimes a drive may be configured for an IP Address, and the drive needs to be taken off-line, and bench tested, or otherwise used outside of its saved IP Settings.

If IP.MODE has been set to 1 (using software defined static IP), the drive will boot up on an IP Address that may be unreachable with the host computer’s settings.

If the IP address prevents communication, the IP settings can be reset to default by the following procedure:

- Set both rotary switches to 0
- Hold down button B1 (top-side of drive) for 5 seconds.

The display will flash 0.0.0.0 and then attempt to discover an address by DHCP.

Without removing logic power from the drive, use WorkBench to connect to the drive, reconfigure the IP address settings as desired, and store the values to non-volatile memory.

4.4 Confirm Connection with the Drive

If you have more than one drive connected to your network, then you can confirm that the new drive is connected to the correct network as follows:

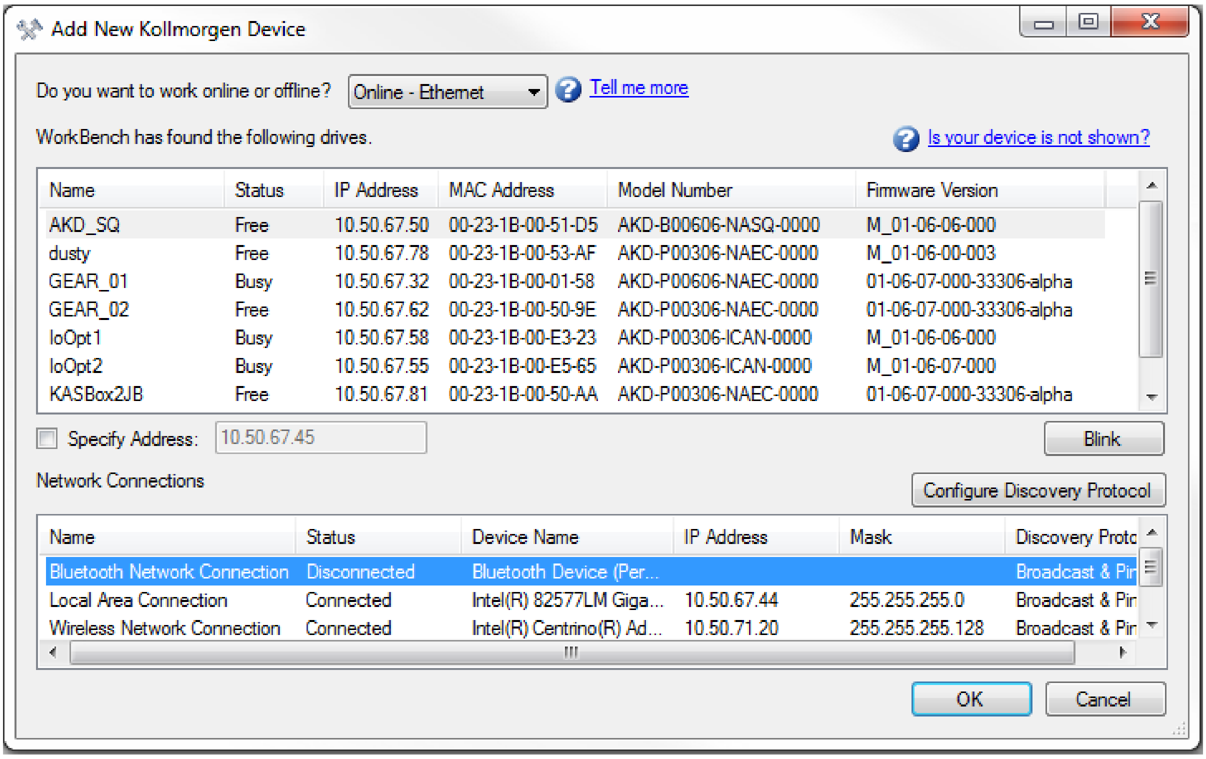
1. A two-digit, seven-segment LED display is located on the front of the drive near the top. If you can see the display, then press the **Blink Display** button on the drive and drive will flash the

seven-segment display on and off.

2. If it is difficult to see the display, then you can check the MAC address in the WorkBench list against the MAC address on the label of the drive. The drive is connected if the numbers displayed in WorkBench match the numbers printed on the label on the side of the drive.

4.5 Connect To Another Drive

Click on Add New Kollmorgen Device... in the bottom left corner of WorkBench to bring up the following window:



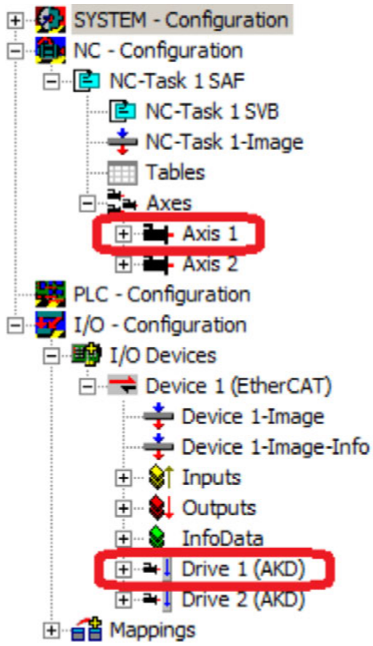
This window allows you to change the drive that you are using.

Button or Dialog Box	Description
Name	Displays the drive name. By default the name is "No_Name". You can change the name by connecting to the drive and navigating to the top item in the navigation tree.
Status	Only one user at a time can connect to a drive. If someone else is connected to the drive, it is **Busy**. If **Free** is displayed, then you can connect.
Blink	Clicking **Blink** forces the display on the selected drive to alternate between the whole display being on and the whole display being off for 20 seconds.
MAC Address	Displays the MAC address of the drive. The MAC address is unique and is also printed on the label on the side of the drive.
IP Address	Displays the IP address of the drive. You can enter a raw IP address (1.2.3.4) or a DNS name. You can also specify a port number different from the default (port 23) by appending the IP address (for example, 1.2.3.4:1000).
Specify Address	If your drive does not appear in the list, you can enter its IP address (for example, 1.2.3.4) or a DNS name. You can also specify a port number different from the default (port 23) by appending the IP address (for example, 1.2.3.4:1000).

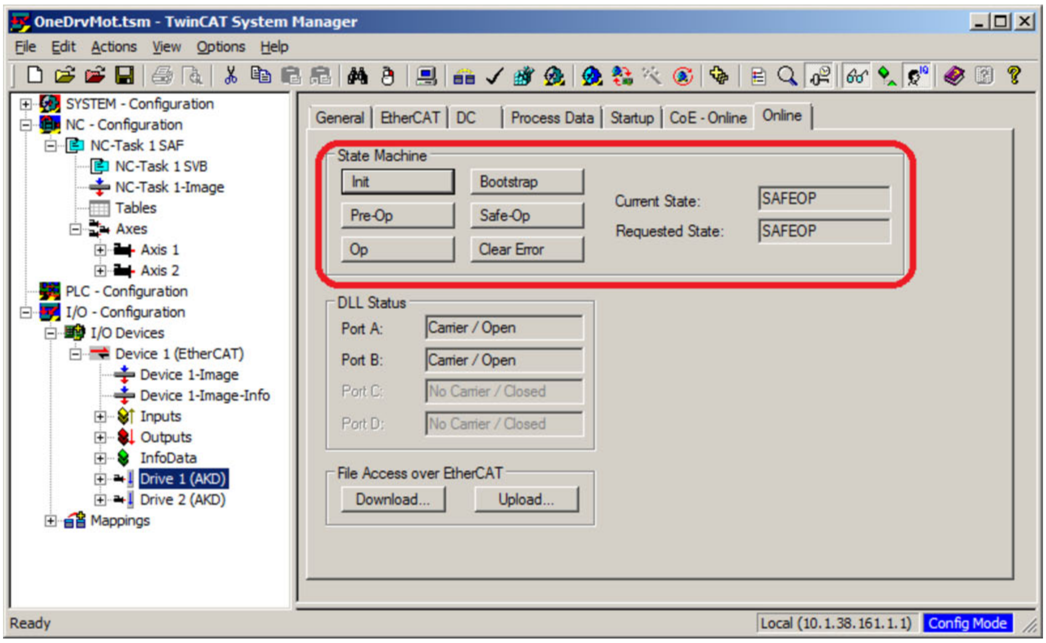
Button or Dialog Box	Description
Configure Discovery Protocol	After selecting a discovery protocol from the list below, you can configure the discovery mode using the following four options: 1. Ping 2. Broadcast 3. Broadcast and Ping 4. No Discovery

4.6 TwinCAT and WorkBenchconfiguration

The EtherCAT network must be setup and managed using TwinCAT System Manager. To be able to connect to a drive and enable it, the drive must be loaded under the I/O Devices node in TwinCAT System Manager and axis must be added to NC - Configuration as shown "Setup via TwinCAT NC/PTP System Manager " in the EtherCAT Manual.



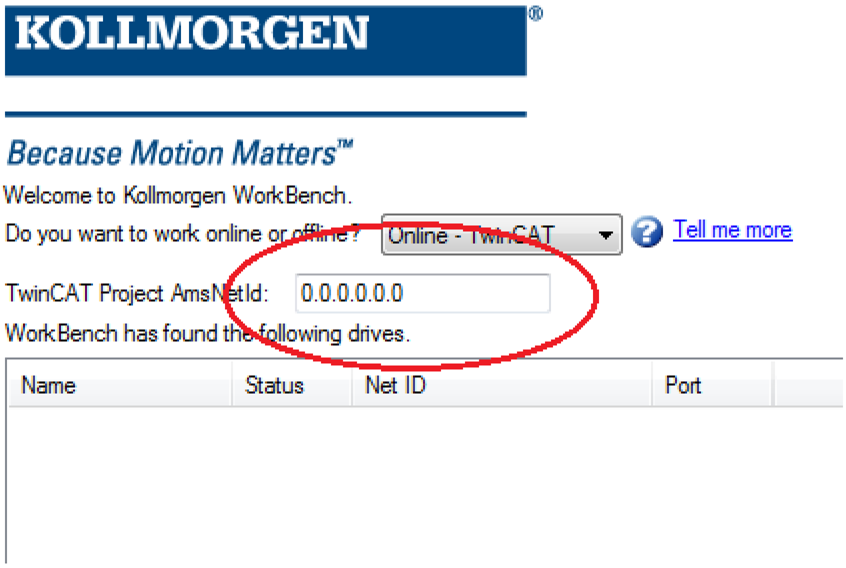
In order to connect to the drives using WorkBench, the drives must be either in Pre-Op, Safe-Op or Op state. State machine for a drive can be accessed from the Online tab for the corresponding drive under the I/O Configuration → I/O Devices → Device [x] → Drive [x] node (see screenshot below).



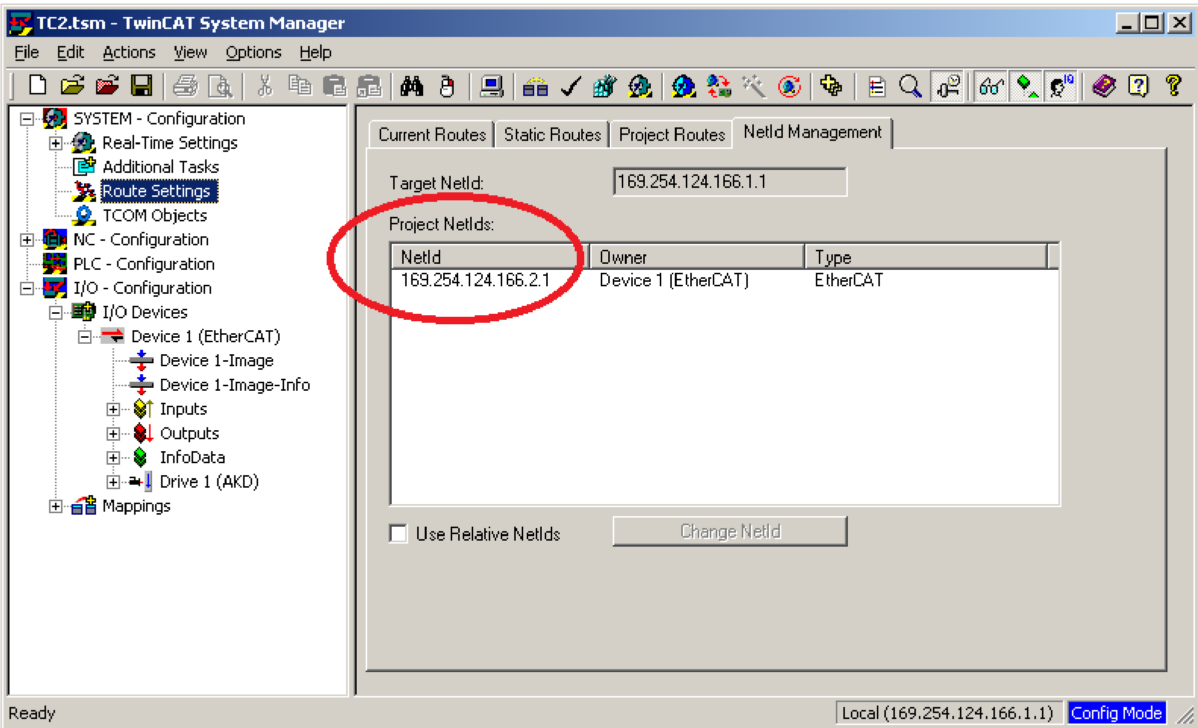
Installation process for WorkBench is the same process as normal, except that it must be installed on the same machine as TwinCAT. Communication to the drive is done thru TwinCAT master and it's not possible to connect WorkBench to the master remotely.

4.6.1 Find and Enter AmsNetId

In order to discover a TwinCAT drive, a valid project AmsNetId must be entered in the field below.



The project AmsNetId can be found in the TwinCAT System Manager under the NetId Management tab in Route Settings. Enter this Id number into WorkBench to discover the drive.



4.7 Troubleshooting Connection and Communication Problems

4.7.1 Device Not Shown

If your specific drive is not shown in the list, then WorkBench has not been able to find the drive.

Common reasons why your drive is not shown in the list include the following:

- The drive is not powered on.
- One of the network cables between your PC and the drive are not connected correctly. You can check if the cable is connected to the drive by checking that the link LED on the Ethernet con-

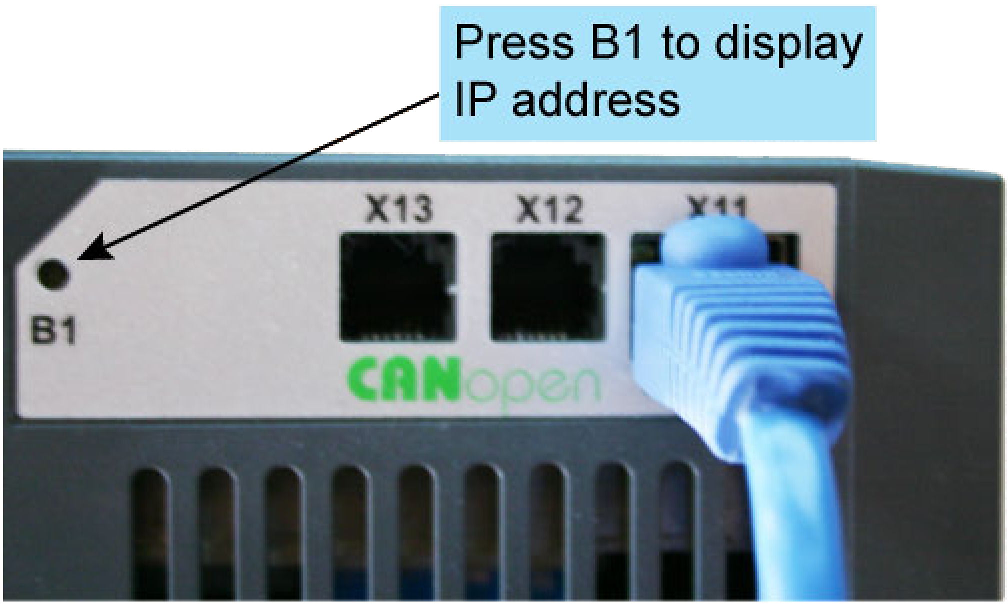
nector is on continuously. If your PC has a link LED, then you should check that this LED is also continuously lit (usually, this LED is next to the RJ45 socket on your PC).

- A router on the network between your PC and the drive is blocking the drive discovery messages. Make sure that port 5002 is not blocked by any routers or firewalls. You can enter the IP address of your drive directly into WorkBench if a router or firewall is blocking port 5002. Often, firewalls are the cause of a blocked connection.
- Your PC and the drive are on different subnets. Networks, especially those with many devices on them, are split up into multiple subnets. The discovery protocol used to find drives will only work if your PC and the drive are both on the same subnet .You can enter the IP address of the drive directly into WorkBench if this is the case.
- The network mask defines more than 512 possible addresses. In this case, WorkBench will not ping all of these addresses, so you must unblock discover ports or specify directly the IP address of your drive. When an adapter has such a network mask, its background will be shown as yellow to warn that this network will not be discovered with all discover protocols.

4.7.2 Find and Enter IP Address

You can view the drive IP address on the drive display by pressing the button shown below. The display shows the digits of the IP address in sequence, with dots separating the numbers. You should see four numbers separated by three dots, for example, 192.168.1.5.

You can enter the IP address if you press **More** and check the **SpecifyAddress** box.



4.7.3 TwinCAT Drive Not Shown

If no TwinCAT device is shown, see Find and Enter AmsNetId (pg 40)

This page intentionally left blank.

5 Communicating with the Drive

5.1 Overview

In order to use the drive, you must be able to communicate with the drive using WorkBench and an Ethernet connection. With some basic network knowledge, you can quickly establish communication with your drive. The drive uses TCP/IP (a worldwide standard for high-speed communication); both the AKD and your PC need to understand each other through this standard in order to communicate. This section explains how to establish a TCP/IP link between your PC and a drive.

5.2 Identifying the Drive IP Address

The first step in establishing communication with the drive is to identify the drive IP address. WorkBench and the drive find each other using this IP address, which tells your PC where to look for the drive in order to make the communication connection. You can establish communication through the IP address with two types of connections:

- **Automatic:** Allow the drive and PC to link automatically.
- **Direct:** Connect to a drive directly based on a known IP address.

NOTE The current IP address can be found at any time by briefly pressing button B1. The address will flash sequentially on the front display.

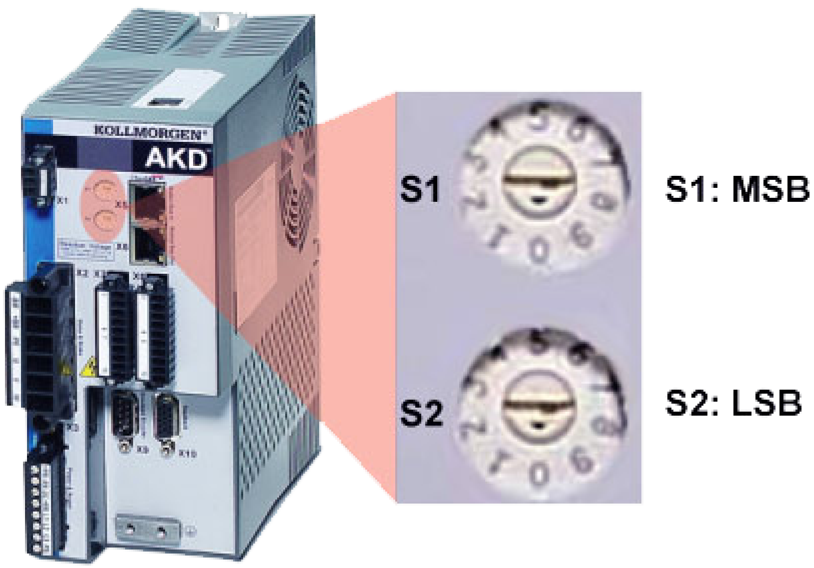
5.2.1 Automatic (Dynamic) IP Addressing

Automatic (also called “dynamic”) addressing is performed using the Dynamic Host Configuration Protocol (DHCP). This protocol makes it easy for a device to attach to a network. The drive is set in automatic IP mode by setting the two rotary switches to zero (S1 and S2, located on the front of the drive). Your PC is set in automatic mode by configuring the TCP/IP screen to "Obtain an IP address automatically"

When first communicating with the drive, conflicts might exist with other programs or devices connected to your computer that are competing for IP addresses. If you have a problem recognizing a drive, then try turning off other devices (especially a wireless device or remote network connection). If you still have problems connecting with the drive, check in the troubleshooting area of this manual.

5.2.2 Static IP Addressing — Rotary Switches

Another option in connecting to the drive is via a static IP connection. In this case you are assigning a specific IP address to the drive and you are modifying your PC network configuration to be able to recognize the static address. The drive IP address can be set using the two rotary switches on the front of the drive.



The address will then be set as 192.168.0.S1S2, with S1 representing the 10’s digit and S2 the 1’s digit. As you turn the switches, the drive displays the S1 and S2 values.

Example:

S1 is set to 3, S2 is set to 5, the address now is set to: 192.168.0.35.

In order for the drive to connect to the PC, the PC network configuration must find this address. First, identify which network port you are using to communicate with the drive. Once you have identified the port, you can access the properties area of the network connection (on your PC) and set up the proper masking to allow the two devices to communicate. The configuration is set up in the “Use the following IP address:”. Set the IP address to 192.168.0.100 and the Subnet mask to 255.255.255.0. This allows the two devices to recognize each other and connect point to point (note that S1 = 0 and S2 = 0 is automatic (dynamic) IP adressing).

Rotary Switch Functions

The following rotary switch settings are used to perform specific functions. After setting the rotary switches, hold down the B1 button on the top of the drive for 5 seconds.

S1	S2	Function
0	0	Resets IP address
1	0	Load drive state from SD card to AKD
1	1	Save current drive state to the SD card.
1	2	Stops BASIC program execution in the drive.
1	3	Starts/restarts BASIC program in the drive.
8	9	Switches DRV.TYPE between EtherCAT and CAN (See CAN-Bus activation with -CC models in the CANopen manual).
9	0	Sets baudrate to auto (See Baudrate for CAN-Bus in the CANopen manual).
9	1	Sets baudrate to 125 (See Baudrate for CAN-Bus in the CANopen manual).
9	2	Sets baudrate to 250 (See Baudrate for CAN-Bus in the CANopen manual).
9	3	Sets baudrate to 500 (See Baudrate for CAN-Bus in the CANopen manual).
9	4	Sets baudrate to 1000 (See Baudrate for CAN-Bus in the CANopen manual).

5.2.3 Static IP Addressing — Software Assigned

Full IP Addressing can be accomplished using four keywords accessible using terminal commands:

- IP.MODE – set Mode=1 to set a static IP address. Set Mode=0 to use rotary switches or DHCP.
- IP.ADDRESS – specifies the address of the drive
- IP.SUBNET – specifies the subnet mask that the drive can communicate with
- IP.GATEWAY – specifies the gateway IP address if the drive needs to communicate outside of its specified subnet

Once the IP address has been properly configured using those four keywords, the IP.RESET command must be issued from the terminal. This will immediately implement the settings that have been configured. These settings must be saved to the drive (DRV.NVSAVE) to remain in effect after power has been removed and restored.

Notes:

- The software assigned static IP address will take precedence over the rotary switches and DHCP.
- To revert back to DHCP settings, or to use the rotary switches to set the address, set IP.MODE=0 and issue the IP.RESET command.

5.2.4 Recovering communications with a drive on an un-reachable IP address

Sometimes a drive may be configured for an IP Address, and the drive needs to be taken off-line, and bench tested, or otherwise used outside of its saved IP Settings. If IP.MODE has been set to 1 (using software defined static IP) the drive will boot up on an IP Address that may be unreachable with the host computer's settings.

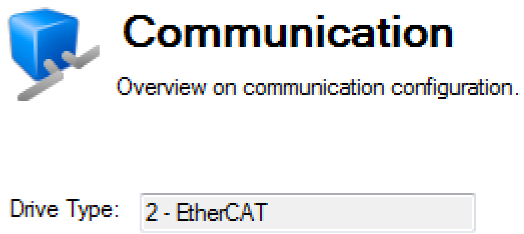
If the IP address prevents communication, the IP settings can be reset to default by the following procedure:

- Set both rotary switches to 0
- Hold down button B1 (top-side of drive) for 5 seconds.

The display will flash 0.0.0.0 and then attempt to discover an address by DHCP. Without removing logic power from the drive, use WorkBench to connect to the drive, reconfigure the IP address settings as desired, and store the values to non-volatile memory.

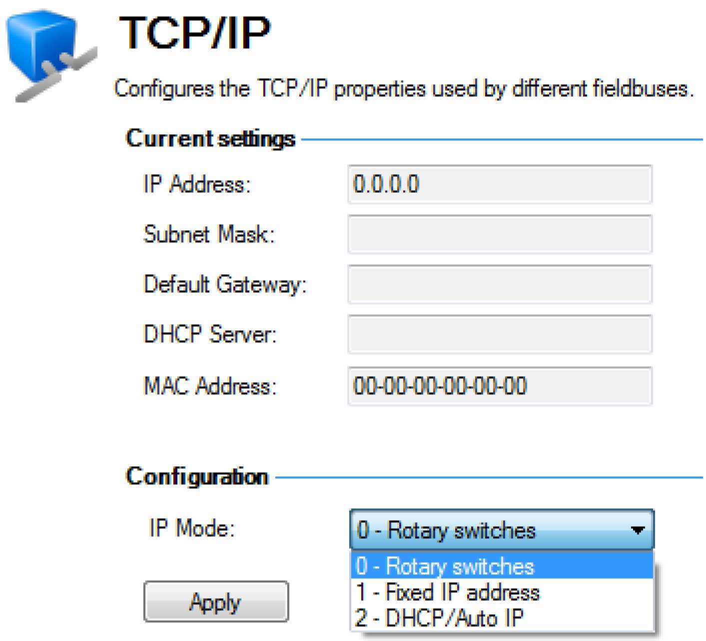
5.3 Communication View

Once the drive is connected, the Communication View will display the drive type as seen below.



5.3.1 TCP/IP View

This view allows the configuration of TCP/IP properties by selecting the IP Mode from the drop-down menu:



5.3.1.1 TCP/IP Communication Protocols

IP Address

The IP address of a drive uniquely defines the drive on the network. Ethernet requires that every device on a network segment must have a unique IP address.

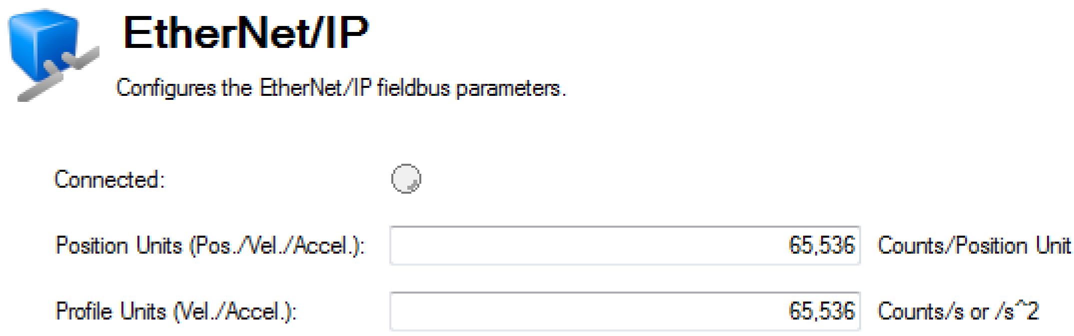
MAC Address

Ethernet also requires that every device must have a globally unique identifier called the MAC address. The MAC address is a 48 bit number normally shown as a series of six hexadecimal numbers (for example, 00:AA:11:BB:22:CC).

Every AKD drive is given a unique MAC address when it is manufactured and this MAC address cannot be changed. The MAC address of every drive is printed on the sticker on the side of the drive.

5.3.2 EtherNet/IP View

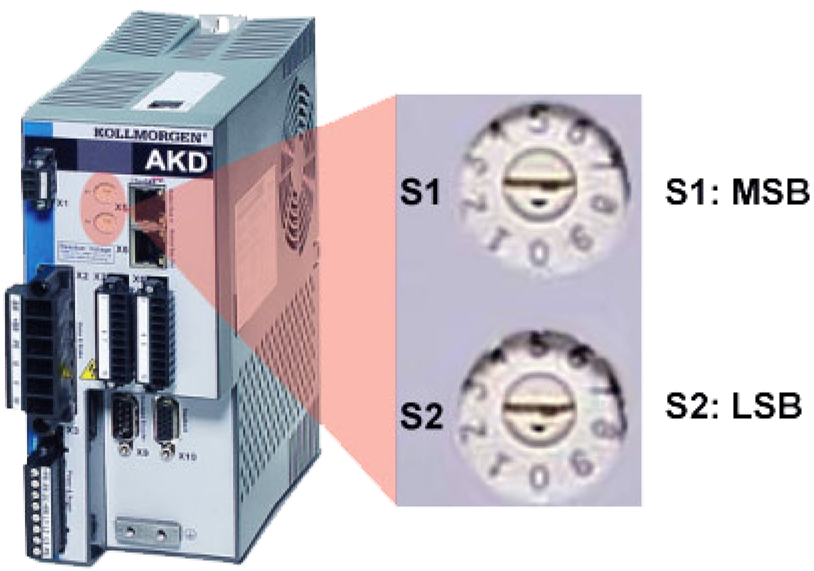
The EtherNet/IP view is only available if connected to an EtherNet/IP drive. From this view you can set position units (EIP.POSUNIT (pg 568)) and profile units (EIP.PROFUNIT (pg 569)). You can also view the state of your connection (EIP.CONNECTED (pg 567)).



5.4 Rotary Switches

5.4.1 Overview

The rotary switches on the AKD are used for configuring the drive without the need for a GUI:



5.4.2 Rotary Switch functions

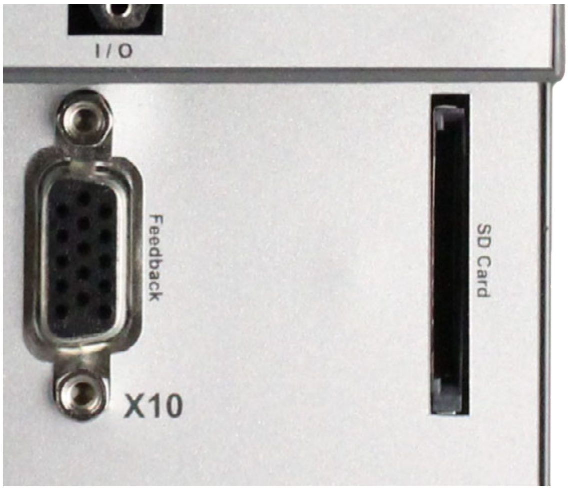
- IP/Network Addressing (refer to)
- Changing CAN-Bus baud rate (refer to)
- Changing Drive Mode (NACC and NBCC models only). Refer to .

- Saving and loading data to the SD Card (AKDs equipped with IO Option Card only). Refer to SD.LOAD (pg 875) and SD.SAVE (pg 876) keyword references.

5.5 Using the SD Card

5.5.1 Overview

The SD Card slot on the AKD is used for backing up and transferring the parameter file and the compiled program file of a drive. This feature is only available on AKD's with I/O option cards.



5.5.2 Save/Load with the SD Card

The following methods are available for saving data to the SD Card, or loading data from the SD Card to drive. Confirm that an SD Card is inserted into the drive slot before performing any of these functions.

NOTE For the drive to recognize and load the parameter file and compiled program file, these files must be named **drive.akd** and **program.bin**, respectively.

5.5.2.1 Save/Load from the Parameter Load/Save Screen:

From WorkBench select the Parameter Load/Save screen from the navigation tree of the appropriate drive. Select either **Load From SD Card...** or **Save To SD Card...** from the list of options

5.5.2.2 Save/Load using SD.SAVE and SD.LOAD:

Call SD.SAVE (pg 876) or SD.LOAD (pg 875) from the WorkBench terminal to execute either command. These commands can only be executed when the drive is in the idle state (i.e. a program is not running) and the drive is disabled.

5.5.2.3 Save/Load using Rotary Switches:

If a computer is not connected to the drive, the save or load command can also be issued using the Rotary Switches (pg 47) S1 and S2. Execute either command by selecting the desired rotary switch settings from the table below, and holding down B1 for 5 seconds.

S1	S2	Function
1	0	Load drive state from SD card to AKD
1	1	Save current drive state to the SD card.

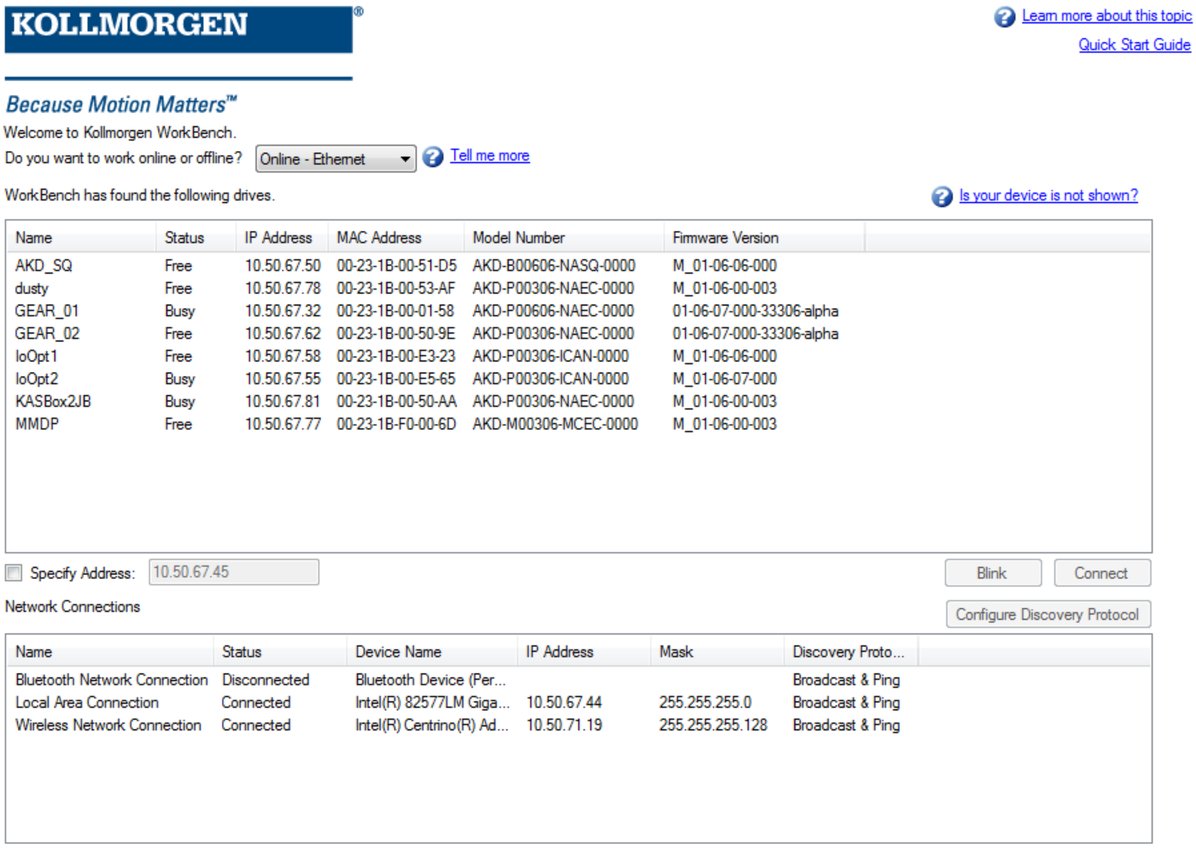
5.5.2.4 Save from PC

If you have drive files on a PC, but do not wish to connect to a drive in order to save them to the SD card, you can also save the files using the PC's resident SD card slot, or an external card reader. Note that for a drive to recognize and load the parameter file and compiled program file, these files must be named **drive.akd** and **program.bin**, respectively.

6 Using WorkBench

6.1 Welcome Screen

This view lets you select which AKD drive you wish to work with. You can work with a physical drive via the Ethernet port of your PC (online) or with a drive simulation (offline).



6.2 Online

Select **Online** to display a list of the AKD drives that WorkBench has found on your local network. You can select one of these drives from the list and click **Next** to continue. This will connect you to the drive and you will be given the option to use a wizard to setup the drive.

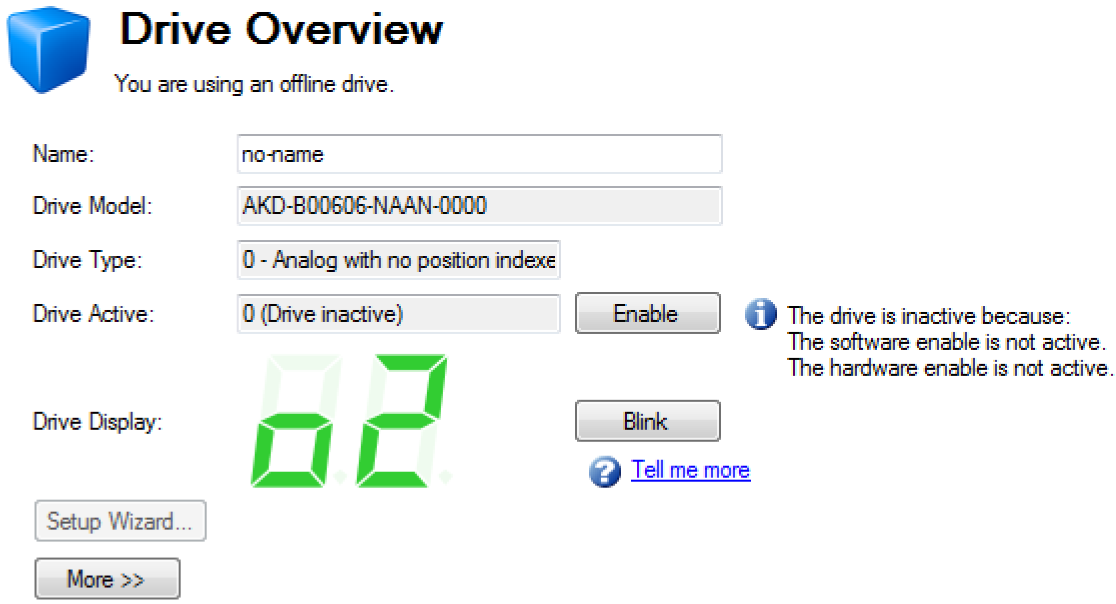
Button or Dialog Box	Description
Name	The name that someone has given the drive. By default, the name is "No_Name". You can change the name by connecting to the drive and navigating to the top item in the navigation tree.
Status	Only one user can connect to an AKD at a time. If someone else is connected to the drive, then the status is **Busy**. If no one is connected to the drive, then the status is **Free** and you will be able to connect.
Blink	Clicking **Blink** will force the display on the selected drive to to repeatadely flash the display LEDs.
MAC Address	This is the MAC address of the drive. The MAC address is unique and is also printed on the label on the side of the drive.
IP Address	This is the IP address of the drive.
Specify Address	If your drive does not appear in the list, you can enter its IP address (e.g. 1.2.3.4) or a DNS name. You can optionally specify a port number different than the default port 23 by appending it. For example, 1.2.3.4:1000 would be port 1,000.

6.3 Offline

Select **Offline** to display a setup screen for the different models that WorkBench can simulate. Once you have made your selection, click **Create** and the **Overview** screen for the Offline drive opens.

6.4 Drive Overview

Once your drive is connected, the **Drive Overview** shows a summary of the drive that you are using.



You can view or edit the following information from the Overview window.

Button or Dialog Box	Description	Parameter
Name	Names each drive in use with a unique identifier.	DRV.NAME (pg 543)
Drive Model	Displays the model number of this drive. The model number is also on the label on the side of the drive. If you are offline, then you can change the type of drive that you are simulating.	DRV.INFO (pg 533)
Drive Type	Selects the operational fieldbus for your drive.	DRV.TYPE (pg 559)
Drive Active	The drive is active when it is enabled and also supplies voltage to the motor.	DRV.ACTIVE (pg 491)
Enable	Click **Enable** to turn on the power stage in the drive and apply voltage to the motor. This command may fail for many reasons; see "DRV.EN " (➜ p. 520) for further details.	DRV.EN (pg 520)
Disable	Click Disable to turn off the power stage and remove the voltage applied to the motor.	DRV.DIS (pg 506)
Drive Display	This graphic replicates the two-digit seven-segment display located on the front face of the drive. The seven-segment display shows a code that indicates the state of the drive and any faults that may be present. WorkBench shows a copy of what the drive display currently shows. A key to the display is here.	

Button or Dialog Box	Description	Parameter
Blink	Click Blink to force the display to alternate between the whole display being on and the whole display being off for 20 seconds. You can use this button to confirm that you are communicating with the correct drive hardware.	DRV.BLINKDISPLAY (pg 492)
Setup Wizard	The Setup wizard takes you through the essential configuration steps so that you can control the motor movement.	
More	Click **More** to display **Serial Number**, **Firmware Version**, **Cumulative On Time**, and **Update Firmware**.	
Serial Number	This text box displays the unique serial number of the drive you are communicating with. The serial number is also shown on the label on the side of the drive.	DRV.INFO (pg 533)
Firmware Version	This text box displays the version of the firmware code running inside the drive.	DRV.VER (pg 561)
Download	Click **Download** to retrieve the latest AKD firmware from Kollmorgen. See Downloading Firmware (pg 280)	
Cumulative On Time	This text box displays the cumulative time this drive has been powered on. When the drive is powered on, this value continues counting from the value it had when the drive was last turned off.	DRV.RUNTIME (pg 553)
Update Firmware	Use this box to select the firmware version you want your drive to run.	

6.4.1 Online and Offline

WorkBench allows you to work online (working with a real drive) or offline (working without any drive hardware).

6.4.1.1 Online Drive

An "online drive" means that WorkBench is working with a specific physical drive on your network.

Each online drive can either be connected to WorkBench (WorkBench has an active connection with the drive and data is being passed between WorkBench and the drive) or it can be disconnected (there is no communication between WorkBench and the drive). If communications are lost (for example a network cable is disconnected) with a drive then WorkBench will switch the drive to the disconnected state.

Only one PC can be connected to a drive at a time.

6.4.1.2 Offline Drive

An offline drive allows you to use WorkBench without having any drive hardware. The parameters of a drive are simulated within WorkBench. An offline drive allows you to create a drive configuration as well as explore the different screens within WorkBench. Because this is a simulation, there are a number of operations that are not possible (for example commanding motion).

6.4.1.3 Switching Between Online and Offline

WorkBench does not allow you to change a drive instance from offline or online. If you wish to move a configuration between two devices that you can save the parameters to a file and then import this parameter file into a new drive you have created.

6.5 Watch

This window allows you to view the current value of selected information from the drive. You can toggle the window on/off by clicking the glasses on the tool bar .

The watch window displays at the bottom of the WorkBench screen. By default, the watch window shows position, velocity, and current for the selected drive. The list can be customized as shown below.

Button or Dialog Box	Description
Add	Adds a new parameter to the watch list.
Edit	Allows you to modify the selected item.
Delete	Removes the selected items from the list.
Move Up	Moves the selected items up one place in the list.
Move Down	Moves the selected items down one place in the list.

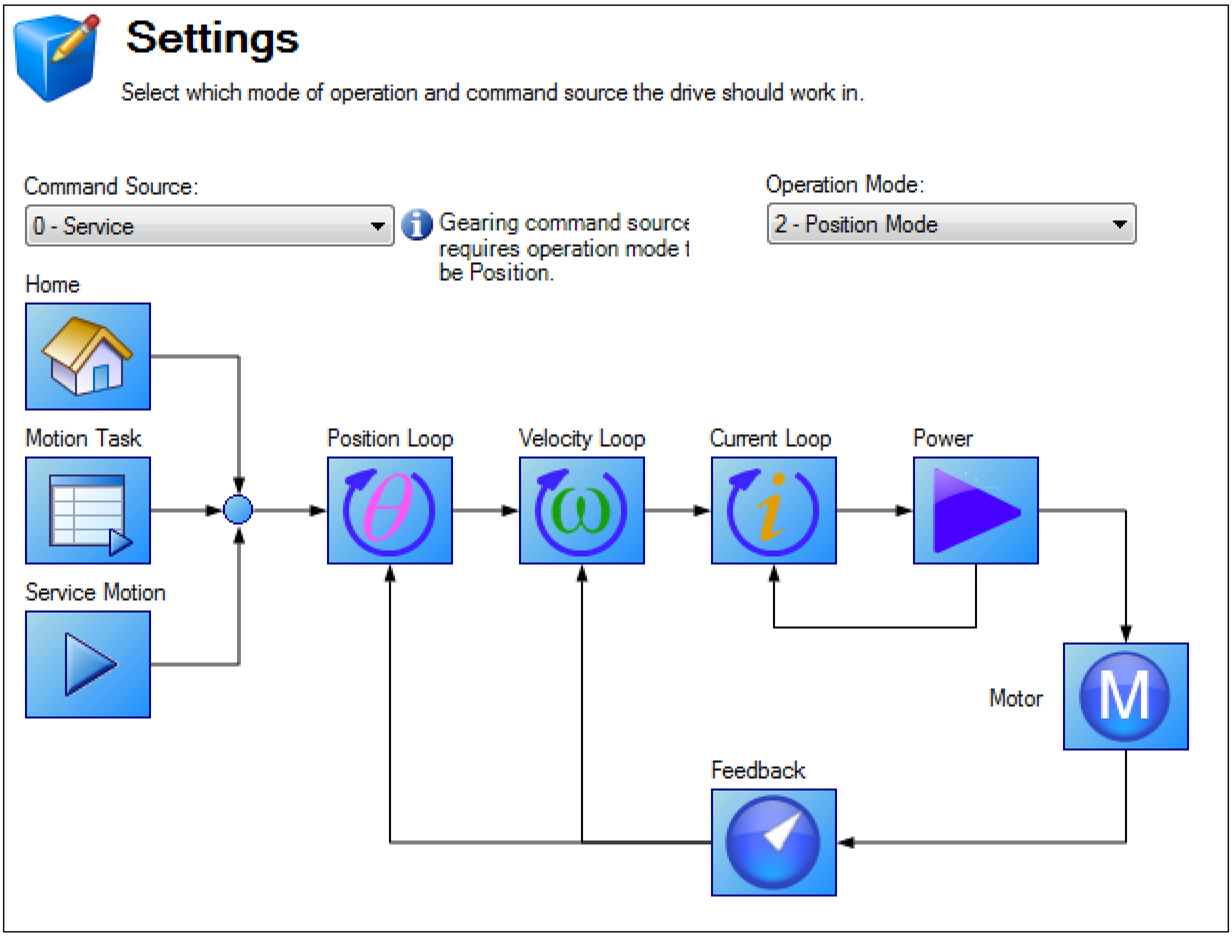
6.6 Settings

6.6.1 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 WorkBench 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 (even if the these settings are not used with the current operation mode and command source), right click on **Settings** and select **Show all Settings**.

6.6.2 Settings View

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



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.
Operation Mode	Selects the control loop being commanded from the source: 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. 1-Velocity Mode: Drive controls are based on the velocity of the motor. 2-Position Mode: Drive controls are based on the position of the motor.
Motion and Control Loop Graphics	Allows you to select the details related to each specific loop from a graphical interface.

Related Topics

Using Command Source and Operating Modes (pg 132) for details on configuring the drive for your application.

7 Configuring Drive Power

7.1 Power

7.1.1 Drive Setup for Power and Bus

The **Power** screen allows you to confirm Power Bus settings and accommodates external regeneration needs if required. Nothing is required for this screen if you have no regeneration requirements. 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 undervoltage fault mode to trigger either only when the drive is enabled or always.

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 (pg 919)
Over Voltage Fault Level	Reads the over voltage fault level.	VBUS.OVFTHRESH (pg 913)
Under Voltage Fault Level	Reads the under voltage fault level.	VBUS.UVFTHRESH (pg 916)
Under Voltage Fault Mode	Sets under voltage mode.	VBUS.UVMODE (pg 917)
Operating Volt-age	Sets the operating voltage.	VBUS.HALFVOLT (pg 912)
Regen Resistor Type	Sets the regen resistor type to either **-1-External Regen** or **0-Internal Regen (if available)**.	REGEN.TYPE (pg 872)
Regen Power	Reads the regen power (only visible for external regen).	REGEN.POWER (pg 867)
External Regen Resistance	Sets the external, user-defined regen resistor resistance (only visible for external regen).	REGEN.REXT (pg 869)
External Regen Heat Up Time	Sets the external regen resistor thermal protection time con-stant (only visible for external regen).	REGEN.TEXT (pg 870)
External Regen Power	Sets the regen resistor's power fault level for an external regen resistor (only visible for external regen).	REGEN.WATTEXT (pg 873)

See Regeneration (pg 59) for more information about regen resistors and sizing regen resistors.

7.1.1.1 Operating Voltage

Operating voltage can be selected by the user to allow AKD-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 (pg 913)).
- 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.

7.1.1.2 Direct DC Mains Operation

Direct DC input is available on all standard AKD 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) (pg 293) for more information on this connection).

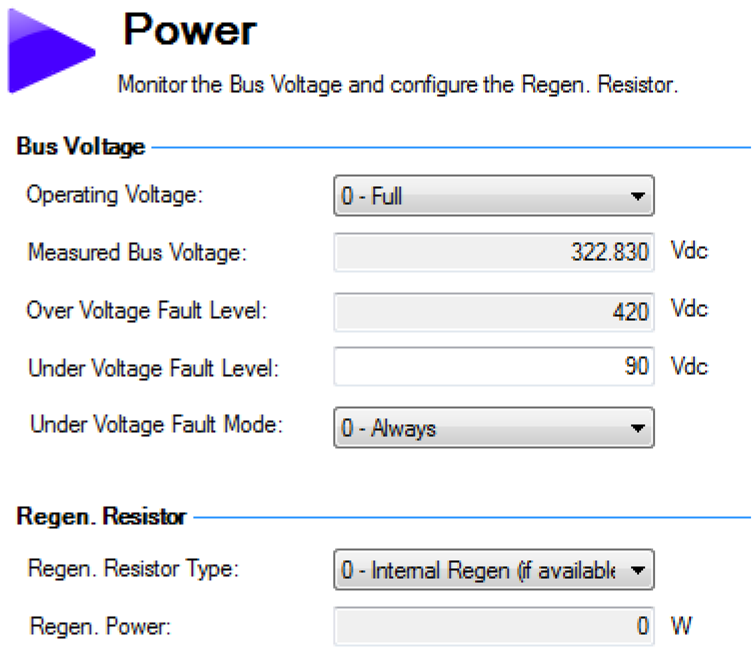
The nominal level of DC voltage applied must be compatible with the voltage fault levels in the drive. You must also consider voltage variations in the DC power supply above and below the nominal value so that nuisance faults are avoided.

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-zzzzz06, 340 Vdc is the equivalent DC voltage for a 240 Vac supply and for the AKD-xxxxx07, 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-zzzz06	90 Vdc	420 Vdc
AKD-zzzz07	380 Vdc	840 Vdc

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



7.2 Regeneration

7.2.1 Overview

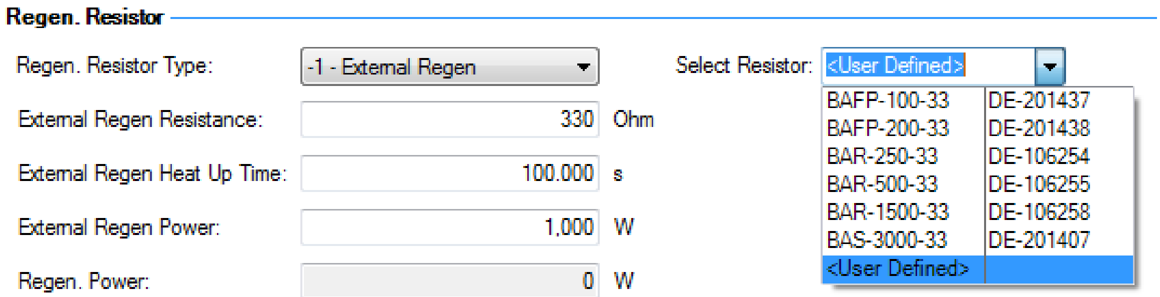
Regeneration, or "shunting", dissipates energy from the DC bus during deceleration of the motor load. During deceleration, the motor acts as a generator that pumps energy back into the system. If this energy is not dissipated, then the bus energy level can exceed acceptable levels (VBUS.OVFTHRESH (pg 913)). If the system exceeds the maximum bus voltage, then the drive generates an over voltage fault (F501) 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 WorkBench.

7.2.2 Regen Resistor Options

In the **Power** screen, you can select from a variety of pre-sized regen resistors using the **Regen Resistor Type** box.



When you select -1-External Regen, the **Select resistor** box appears and contains the 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.

7.2.3 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.

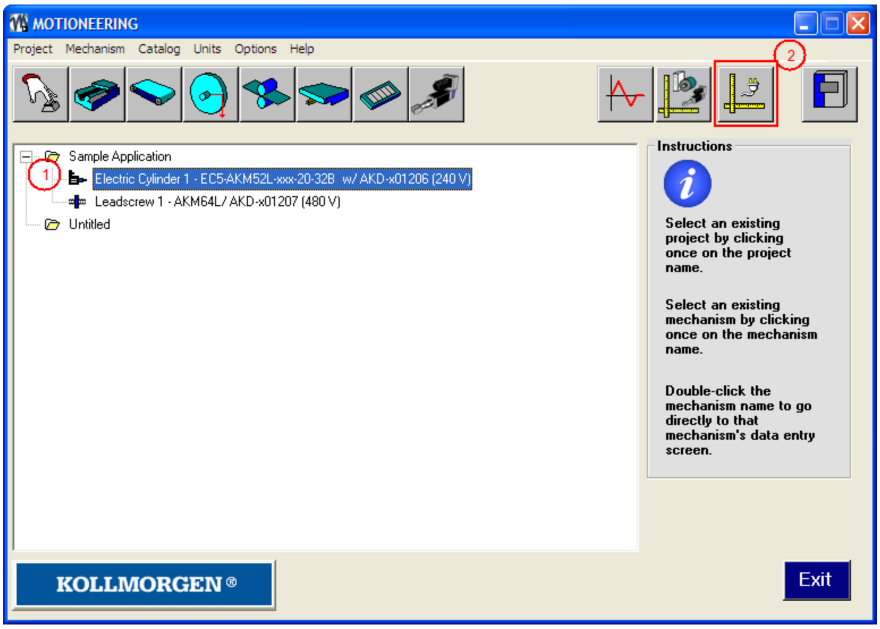
The calculation for peak kinetic energy requires values for several factors that affect the generation of energy in a motion system:

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

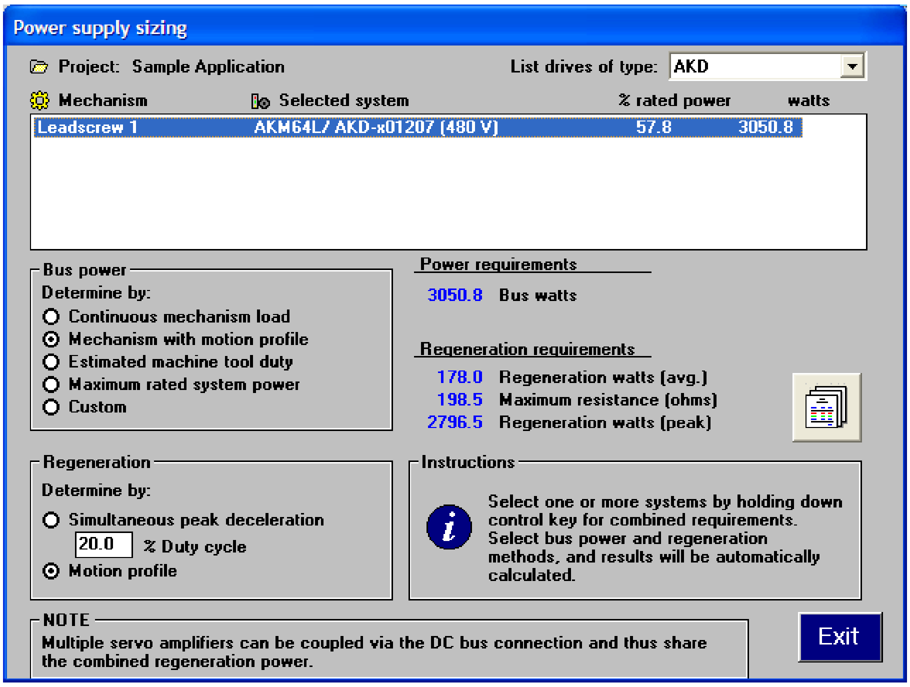
Regen resistor sizing information for your particular application can be calculated using the Motioneering® Application Engine. You can download this program here:

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.



7.2.4 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 WorkBench 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-x-00306	AKD-x-00606	AKD-x-01206	AKD-x-02406	AKD-x-00307	AKD-x-00607	AKD-x-01207	AKD-x-02407
BAFP-100-33	DE-201437	External Resistor, 100 W, 33 ohms	x	x	x					
BAFP-200-33	DE-201438	External Resistor, 200 W, 33 ohms	x	x	x					
BAR-250-33	DE-106254	External Resistor, 250 W, 33 ohms	x	x	x		x	x	x	
BAR-500-33	DE-106255	External Resistor, 500 W, 33 ohms	x	x	x		x	x	x	
BAR-1500-33	DE-106258	External Resistor, 1500 W, 33 ohms	x	x	x		x	x	x	
BAS-3000-33	DE-201407	External Resistor, 3000 W, 33 ohms	x	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
BAR-500-15	DE-201439	External Resistor, 500 W, 15 ohms				x				
BAR-1000-15	DE-201440	External Resistor, 1000 W, 15 ohms				x				
BAS-2000-15	DE-103871	External Resistor, 2000 W, 15 ohms				x				
BAS-3000-15	DE-103872	External Resistor, 3000 W, 15 ohms				x				
BAS-6000-15	DE-103873	External Resistor, 6000 W, 15 ohms				x				

7.2.5 Configuring Regen Parameter Values

If you use an external regen resistor, then you must also include additional information about the regen resistor in the**Power** view. These values are automatically inserted in WorkBench. 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
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 (pg 866)

VBUS.OVWTHRESH (pg 914)

VBUS.VALUE (pg 919)

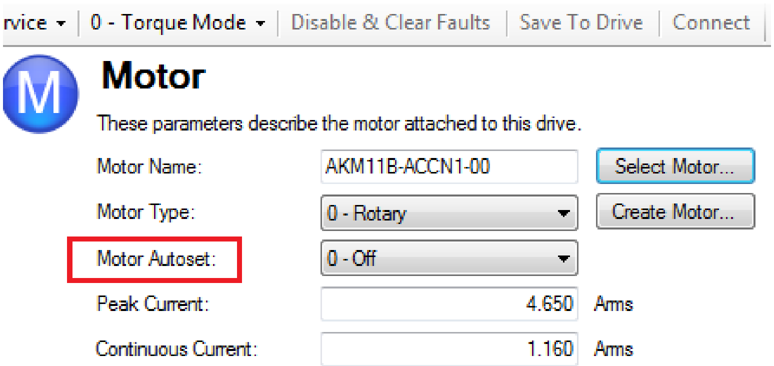
8 Configuring Motor Settings

8.1 Motor

8.1.1 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 autoset feature as shown in the screenshot below (MOTOR.AUTOSET = 0) and select the appropriate motor from the motor parameter database.



All of the appropriate Kollmorgen motors compatible with the AKD 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.

8.1.2 Motor Setup

When motor autoset is on (MOTOR.AUTOSET = 1), the AKD 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 (pg 65) screen.

8.1.3 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 autoset 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).
- Motor Autoset: This setting to allows the drive to automatically set up a plug and play motor (MOTOR.AUTOSET = 1). With **Motor Autoset** turned off (MOTOR.AUTOSET = 0), you can access the motor database to select a catalog or custom motor.

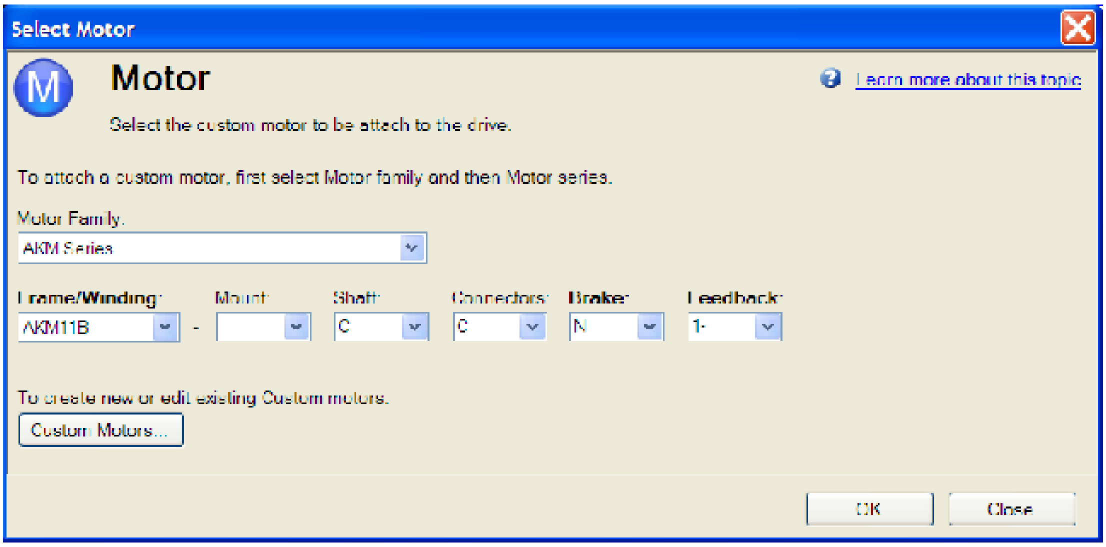
The next set of parameters displayed are specific to the electrical and mechanical characteristics of the motor connected to the AKD drive.

- Peak Current: Motor peak current rating in Amps rms.
- Continuous Current: Motor continuous current rating in Amps rms.
- Inertia: Motor rotor inertia in Kg-cm^2.
- Torque Constant: Motor torque contant in Nm/Arms.
- Inductance: Motor rated inductance in milliHenries
- Motor Poles: Number of motor poles.

- Maximum Speed: Motor maximum rated speed
- Motor Resistance: Motor winding resistance in Ohms.
- Maximum Voltage:Motor maximum rated voltage in Volts rms.
- Motor Phase: Motor phase offset (used to set motor commutation as required - for most devices this is set to 0).
- Coil Thermal Constant: Motor coil thermal time constant in mHz.

8.1.4 Selecting a Motor

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



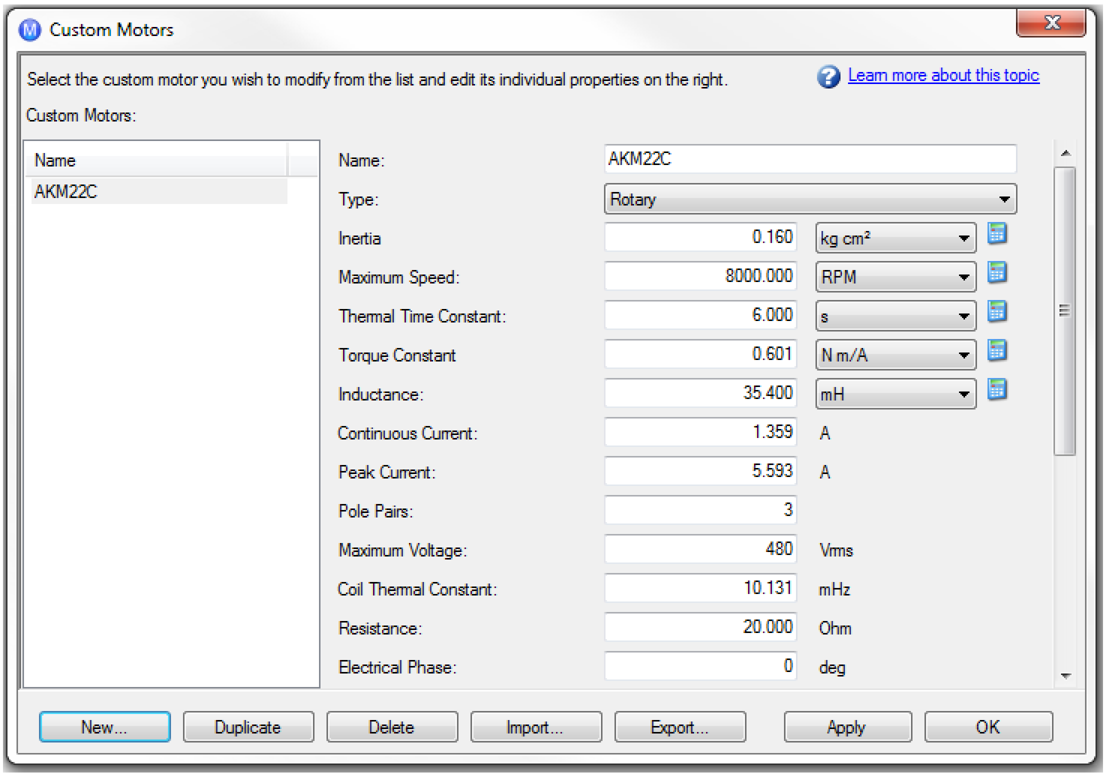
When this screen is opened, WorkBench displays by default the motor matched to the current motor name attached to the drive. WorkBench searches matching motor as follows:

1. WorkBench first checks the motor name with custom motors for a match.
2. If a match is not found, then WorkBench 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.

8.1.5 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 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.

8.1.5.1 Validating Motor Parameters

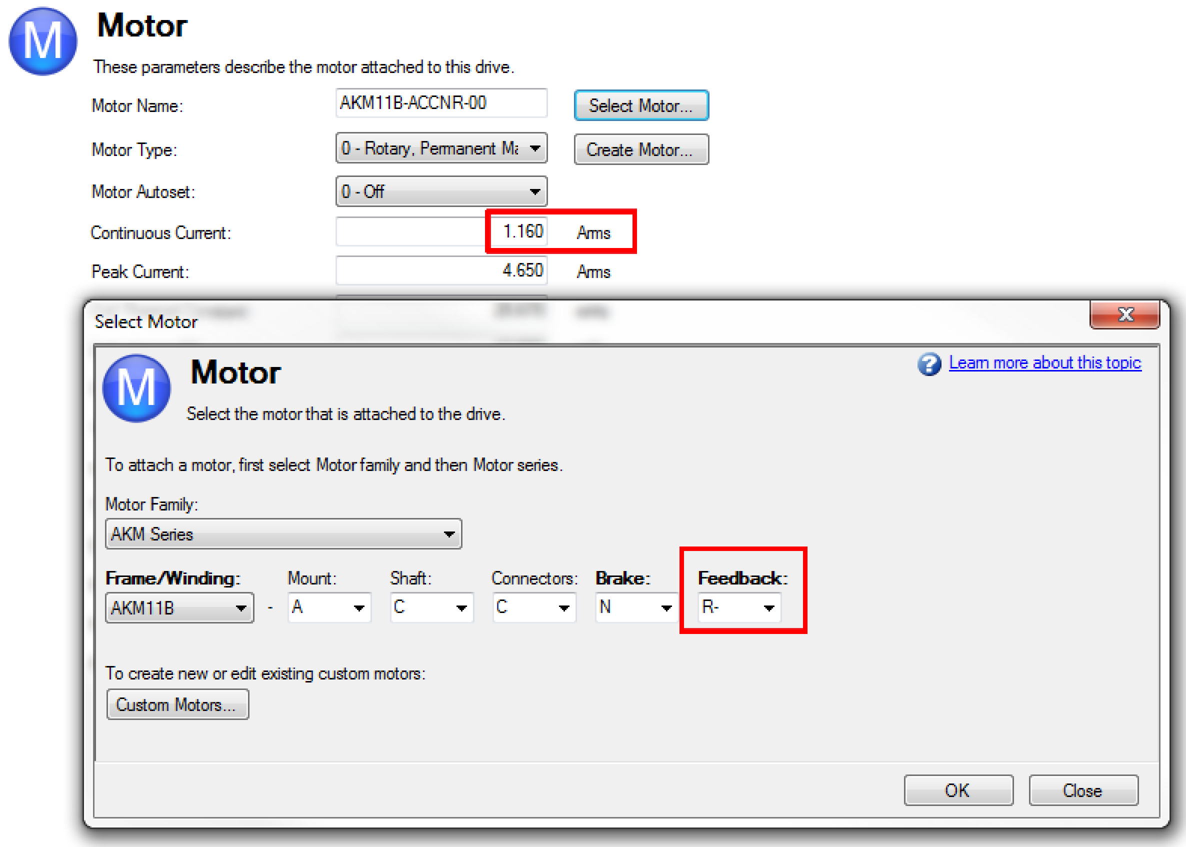
When you click **OK** in the **Select Motor** screen, WorkBench validates the range with the drive. If any error is found, an error screen is displayed. Click **Continue** to set the motor parameters in the drive. Click **Cancel** to close this screen.

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

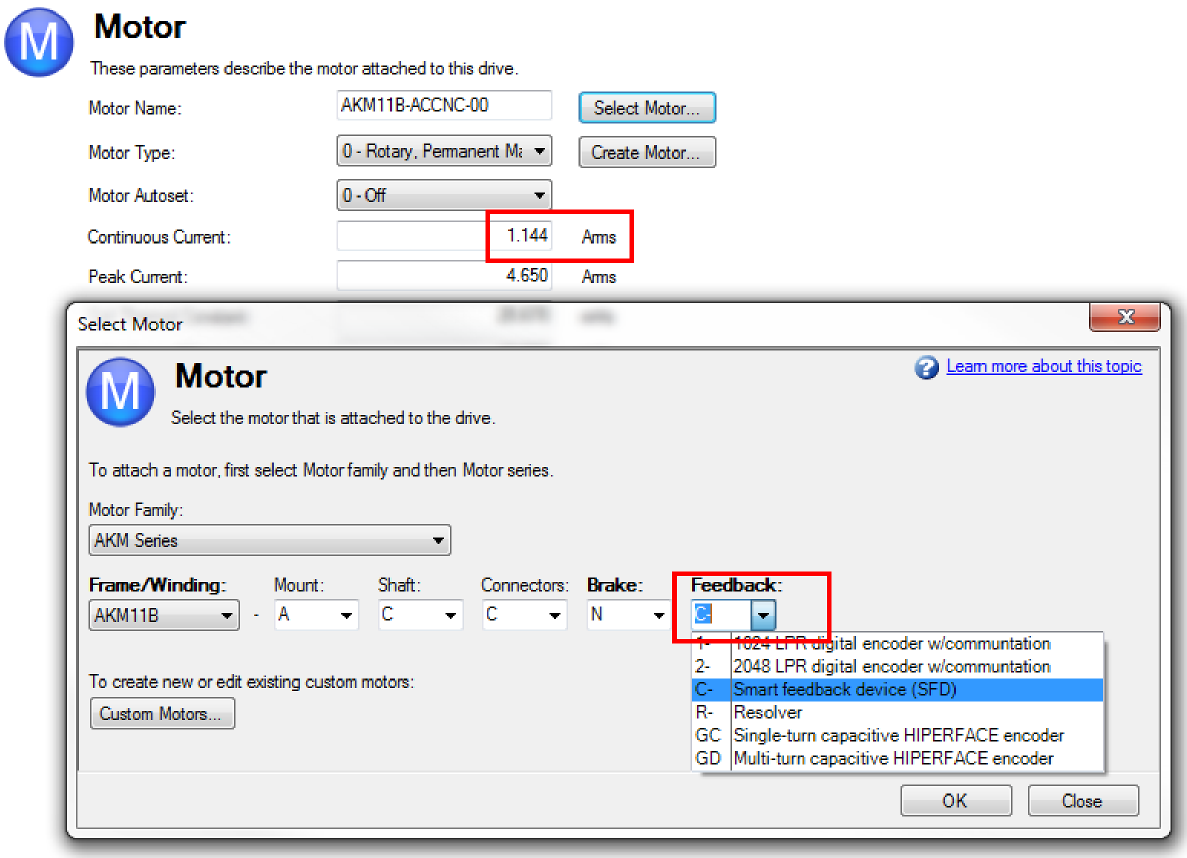
8.1.6 Motor Derating

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):

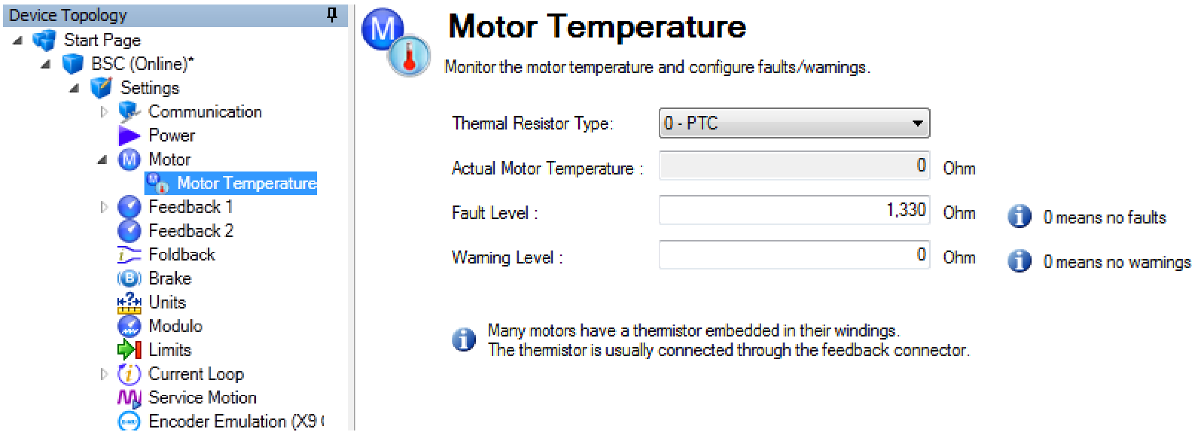


SFD with no brake (derated):



8.1.7 Motor Temperature

The motor temperature settings are located in the motor temperature view.



Related Parameters

MOTOR Parameters (pg 741)

8.2 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.

8.2.1 Overview

The AKD offers a variety of feedback solutions, which allow you to optimize your system based on your specific machine needs. The available feedback options include resolver, SFD, sine-cosine encoder (Endat 2.1, BiSS, Hiperface), incremental encoder, as well as line-count, single, and absolute variations. Your motor model number will indicate the type of feedback that you have. With some incremental encoders, parameters are set up in the encoder itself, and the AKD drive recognizes the feedback automatically and sets up the drive accordingly. This automatic recognition is called "plug and play". Currently, SFD and Endat are plug and play encoders. Other feedback types require that you enter parameters manually.

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

		Primary	Secondary
Resolver	Std & Multi pole	Yes	No
SFD		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	All Digital (Mode C)	Yes	No
	Analog / Digital	Yes	No

		Primary	Secondary
Hiperface	Analog / Digital; Single & Multi Turn	Yes	No
	All Digital	No	No

8.2.2 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.

8.2.2.1 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.

8.2.2.2 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).

8.2.2.3 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.

8.2.2.4 Endat 2.1, Endat 2.2

Endat-based sine are plug and play compatible, and the system will properly recognize these encoders. With the AKD set in **Auto**, this encoder type is detected and the feedback and motor parameters are loaded automatically.

8.2.2.5 BiSS

BiSS will be plug and play in a future release. Currently, the device is programmed with the motor and feedback information and once selected will sets up the feedback and motor parameters in the AKD.

8.2.2.6 Hiperface

Hiperface is a plug and play device that will be supported in a future release.

8.2.2.7 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 does not support non-Kollmorgen standard resolver options.

8.2.2.8 SFD

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.

8.2.3 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 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 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 (pg 962) command over telnet.

The AKD 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 70)

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 (pg 966) (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 75)

Mode 2:

Mode 2 is the easiest of the Wake and Shake modes. It requires no parameters to be configured. 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.

8.2.3.1 Using Wake and Shake Mode 0 (WS.MODE 0)

Overview

Wake and shake (WS) is used to establish commutation in drives with the following types of feedback:

- Incremental encoders without Halls or commutation channels.
- Sine encoders without Halls or commutation channels.

When controlling a brushless DC (BLDC) motor, you must know the electrical position of the motor shaft. Without absolute position data, it is impossible for the drive to know which sequence of coils to energize to produce motion. Absolute feedback devices, such as resolvers and absolute encoders, can detect position directly. Incremental devices, such as incremental encoders and sine encoders without a commutation channel, must determine electrical position indirectly at start up. The drive uses the WS feature to determine electrical position by sending the motor short bursts of current and measuring the resulting incremental motion. The drive uses this measurement to estimate electrical position accurately enough to control the motor.

Configuring WS

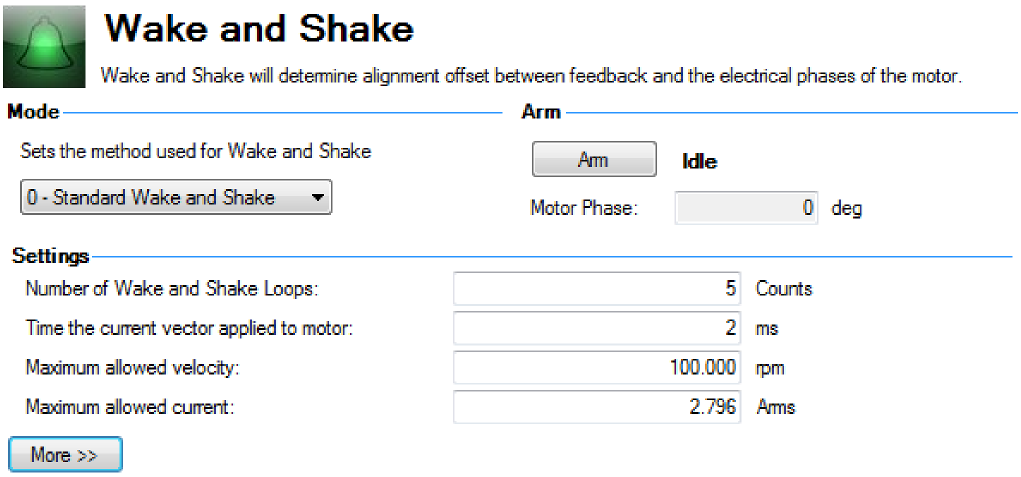
You can configure WS after your motor has been connected to the AKD according to the *AKD Installation Manual*. The WS procedure is initiated automatically when both the hardware and software enable signals become logic high.

Before attempting to enable the drive, the drive must be compensated for the motor and the AKD servo loops must be stable.

Compensation values for many rotary motors are included in a database already loaded into the drive.

NOTE An unstable system will not function properly during or after the WS process.

Use the default **Wake and Shake** view to configure your system:



Arm

Click **Arm** to set WS to start at the next drive enable (WS.ARM (pg 958)). This area also shows the current status of the wake and shake process. See WS.STATE (pg 969) for a detailed explanation of the possible states.

WS.ARM is not restricted to any feedback type.

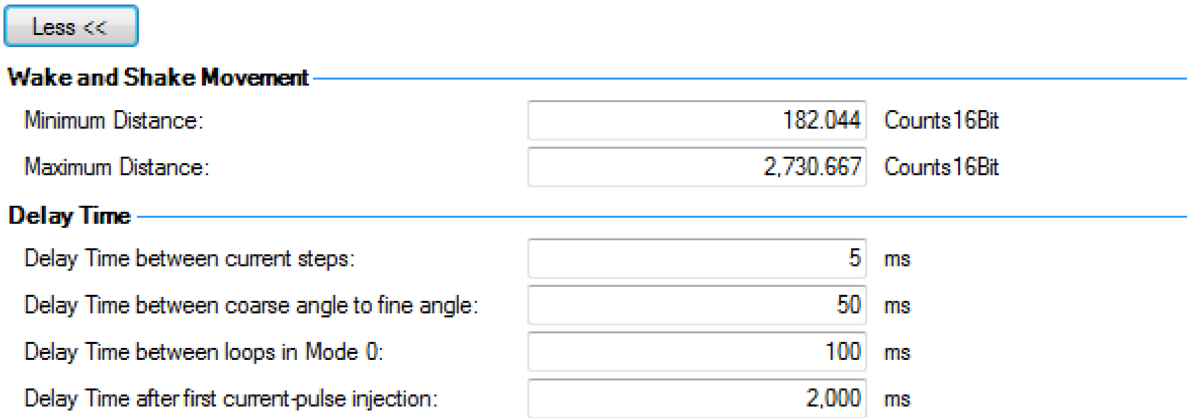
Settings

- **Number of Wake and Shake Loops**. The WS feature uses the mean of all wake and shake repetitions, called "loops", to establish commutation (WS.NUMLOOPS (pg 968); see Using WS: Advanced (pg 73) for a discussion of loops). If fewer than five loops are used, commutation may be incorrect, possibly causing poor performance or stability.
- **Time the current vector applied to motor.** This box specifies the duration of the current pulse used for commutation. Increasing this value (WS.T (pg 970)) increases the movement of the system.
- **Maximum allowed velocity**. If a velocity (VL.FB (pg 934)) higher than this value (WS.VTHRESH (pg 976)) is detected while WS is running, then a fault will be generated.
- **Maximum allowed current**. This value (WS.IMAX (pg 966)) is directly proportional to the movement. A value that is too low may fail to cause movement; a value that is too high value may cause an over speed fault.

- **Maximum allowed movement** (WS.DISTMAX (pg 963)). If the total motion from the starting position (the position at the time the drive is enabled after a WS.ARM command) exceeds WS.DISTMAX a fault will occur. Setting WS.DISTMAX to zero disables this feature.
- **Minimum allowed movement** (WS.DISTMIN). If the total motion from the starting position is less than WS.DISTMIN (pg 964) a fault will occur. This will prevent poor initialization from broken wires, incorrect current settings, very high friction, etc. Setting WS.DISTMIN to zero disables this feature.

Wake and Shake, More View

To configure additional WS settings, click **More** at the bottom of the default view to display the following options:



Wake and Shake Movement

Use these boxes to set values for the maximum (WS.DISTMAX (pg 963)) and minimum (WS.DISTMIN (pg 964)) movement required for finding commutation.

Delay Times

Delay time is the time that elapses when switching different current vectors. Use these boxes to set specific time delays for current steps(WS.TDELAY1 (pg 971)),coarse to fine angle (WS.TDELAY2 (pg 972)) and time between loops in mode 0 (WS.TDELAY3 (pg 973)).

Special Cases for WS

Operation with Motor Brake

An amplifier with a motor brake operates the WS procedure similar to an amplifier without a brake. All precautions and behavioral descriptions above also apply in this case. It is important to note that the brake is automatically applied (motor brake, not holding brake) after the WS process is complete. The brake may cause unexpected movement if the DRV.OPMODE used prior to WS does not retain position. If a force component is present parallel to the track on a linear motor (gravity, load, etc.), or tangential on a rotary motor, the motor may move from the startup position after WS completes and the brake is applied.

If the application requires that the startup position be retained, have the controller system ready to take control immediately after WS is complete. One way to set this control is to have the drive in DRV.OPMODE 1 (digital velocity) or DRV.OPMODE 2 (position mode) on power-up. This precaution keeps the motor stationary after enable.

End of Travel Limits

If anything restricts the motion of the motor, a commutation fault can occur. Examples of situations that may result in faults include the following:

1. If the motor is resting against a rigid end stop, the movement of the motor may be impeded below the minimum threshold set by WS.DISTMIN. This lack of movement causes a fault.

2. If the motor is actuating a limit sensor/switch, the system (PLC, SWLS.LIMIT0 and SWLS.LIMIT1) may be preventing the AKD from producing motion. If descriptive motion is not achieved, the system faults.

Large Load Inertia or High Friction System

Systems with a large load mismatch may need more current than the default setting for correct commutation. Begin with the default value for WS.IMAX and gradually increase or decrease as needed. If adjusting WS.IMAX does not result in a successful commutation, the width of the search pulse can be increased by increasing WS.T.

Using WS: Advanced

WS is performed upon enable in order to establish a valid value for MOTOR.PHASE at startup. MOTOR.PHASE is used to calculate electrical phase. With absolute feedback devices, MOTOR.PHASE is a fixed offset between absolute mechanical position and the electrical position. With incremental devices, position is accumulated relative to an initial MOTOR.PHASE. However, at startup, MOTOR.PHASE is invalid since the initial position is random, thus the requirement for the WS process.

WS is a two-step process:

1. Coarse Phase. The drive sequentially pulses a user-specified current, WS.IMAX (pg 966), at each electrical quadrant (0°, 90°, 180°, 270°). Based on the resulting observed movement, an approximate location is calculated.
2. Fine Phase. The drive makes small adjustments to the coarse phase while monitoring movement during velocity mode (command velocity = 0) to find a precise position.

The amplitude of the current pulses in this process equals WS.IMAX. The drive repeats these two steps for a user-specified number of times (WS.NUMLOOPS) to produce a more accurate estimate of the electrical phase.

The drive normally indicates warning F478 (pg 260) before WS is initiated and successful. If WS fails, the commutation is not valid and the drive indicates one of the following faults:

- F473 (pg 259): Insufficient movement. The maximum movement during WS was less than WS.DISTMIN.
- F475 (pg 260): Excessive movement. The movement during WS exceeded WS.DISTMAX.
- F476 (pg 260): Fine-Coarse deltat too large. The phase calculated during the fine phase and coarse phase differed by more than 10 degrees.
- F478 (pg 260): Overspeed. The feedback velocity (VL.FB) exceeded WS.VTHRESH during WS.
- F479 (pg 260): Loop angle delta too large. The difference between the phase determined in different cycles (loops) exceeded 30 degrees.
- F482 (pg 260): Commutation not initialized. WS is required (feedback is one of the types listed in the Overview) but WS has not been successfully performed.
- F483 (pg 260) to F485 (pg 260): U, V, or W phase missing . Intermittent or broken motor connection.

Maximizing WS Reliability

The following suggestions will help you achieve successful commutation:

- Incorrect determination of MOTOR.PHASE may cause a system runaway. Since the typical movement during correct operation of WS is very small, you can use the velocity overspeed parameters (WS.VTHRESH and DRV.VTHRESH) to prevent a runaway. Prior to enabling the drive, set DRV.VTHRESH 100 mm/s for linear motors or 200 rpm for rotary motors. After a successful enable, DRV.VTHRESH can be returned to the normal operating value.
- Set WS.IMAX to its default value, WS.IMAX=0.5*min(MOTOR.IPEAK, DRV.IPEAK).
- Set WS.NUMLOOPS 20 for best results in many applications.
- WS.T specifies the duration for which the search current is applied. With a stable velocity loop, most applications work well with the default value of WS.T. The default value causes the software to calculate the width of the search pulse based on the velocity loop proportional gain, VL.KP.

Increasing WS.T effectively increases the movement of the motor during WS, which may be necessary for systems with a low-resolution feedback or high load inertia.

- WS.IMAX specifies the amplitude of the current pulse used during the initial/rough commutation. A WS.IMAX value that is too low may result in a fault by failing to cause enough movement for commutation. If the value is too high, the preset movement threshold could be exceeded, also resulting in a fault. If the default value is producing faults for too little movement, gradually adjust this parameter to overcome excessive friction and/or load on the system. WS.IMAX also specifies the maximum current used in the second stage of commutation. The initial current is 25% of WS.IMAX, then steps up to 100% of WS.IMAX.
- FB1.SELECT selects the type of feedback used by the amplifier. The WS feature is used only for FB1.SELECT = 11, 21. The feedback must be configured prior to initiating the WS procedure.
- If your amplifier has a motor brake, set MOTOR.BRAKE = 1. For motors without a brake, set MOTOR.BRAKE = 0.

⚠CAUTION

- Adjust WS.T with extreme caution. Increasing WS.T increases the movement of the system. Applying an incorrect value of WS.T may cause erratic drive behavior.
- When initiating WS, the motor may experience a runaway. Stay clear of all moving parts. Ensure there are properly operating safety devices such as hardware limit switches and suitable end-of-travel limits.
- If WS.NUMLOOPS is less than 5, commutation may be incorrect. This condition may cause faults and/or adversely affect performance or stability. Set WS.NUMLOOPS = 20.

Troubleshooting WS

Problem	Possible Cause	Remedy
Excessive Movement	• Brake slips on vertical system. • External forces on motor too great.	• Check brake. • Remove forces acting on motor. • Lower WS.IMAX.
Insufficient Movement	• Motor brake too rigid. • Motor resting on rigid end-stop. • Too much friction on motor track. • Foreign objects impeding movement of motor. • Motor load very large, and impedes sufficient movement. • WS.DISTMIN manually set too high • WS.IMAX too low	a. Check brake b. Check physical location of motor c. Check friction and cleanliness of motor track d. Increase WS.IMAX value
Excessive Movement	• Brake slips on vertical system. • External forces on motor too great.	• Check brake. • Remove forces acting on motor. • Lower WS.IMAX.

Problem	Possible Cause	Remedy
Insufficient Movement	• Motor brake too rigid. • Motor resting on rigid end-stop. • Too much friction on motor track. • Foreign objects impeding movement of motor. • Motor load very large, and impedes sufficient movement. WS.DISTMIN manually set too high WS.IMAX too low	• Check brake. • Check physical location of motor. • Check friction and cleanliness of motor track Increase WS.IMAX value
U, V, or W Phase Missing Fault	Intermittent or broken motor connection.	Check connections to motor phases.
Commutation Not Initialized Fault	Wake and Shake is required but WS procedure has previously been canceled (WS.DISARM) or has failed.	Correct errors and rerun WS procedure.
Other	• Drive not configured correctly.	• Check drive compensation • Verify amplifier feedback

Related Parameters

WS Parameters (pg 957)

DRV.IPEAK (pg 535)

FB1.SELECT (pg 603)

MOTOR.BRAKE (pg 744)

MOTOR.PHASE (pg 760)

MOTOR.IPEAK (pg 755)

8.2.3.2 Using Wake and Shake Mode 1 (WS.MODE 1)

Overview

Wake and Shake mode 1 uses a pole locking method (sets the motor pole count to 0) and applying current to determine the commutation angle.

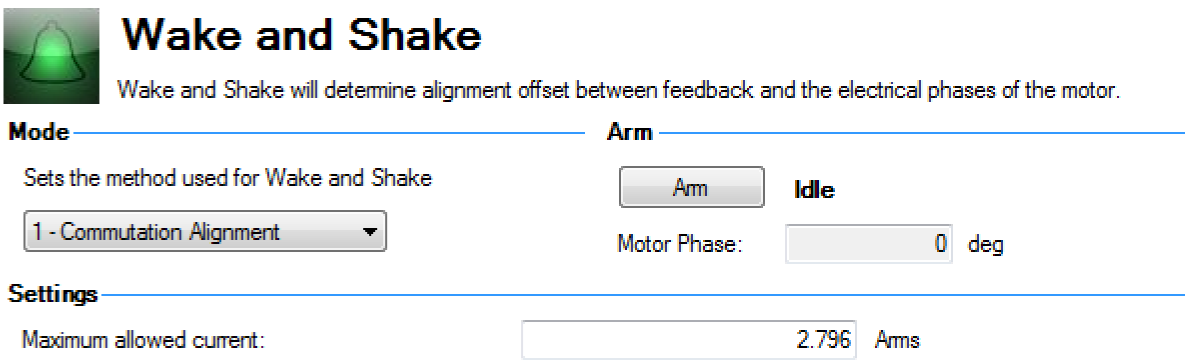
Wake and shake is required for the following applications:

- Incremental encoders without Halls or commutation channels.
- Sine encoders without Halls or commutation channels.
- Motor with Halls which were not aligned to be at 0 degrees motor phase.

Configuring Wake and Shake

Wake and Shake mode 1 has two configurable parameters: WS.IMAX and WS.TSTANDSTILL.

WS.IMAX (pg 966) can be configured from the Wake and Shake view.

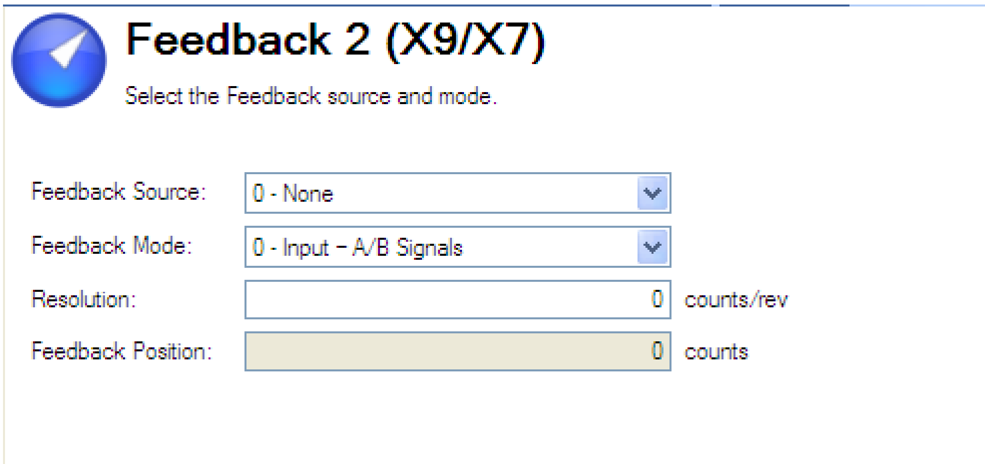


WS.IMAX is the value (in amps) of the current applied to the motor to lock the motor to a magnetic pole position. This value should be enough to force the motor to move. But if the motor oscillates strongly during Wake and Shake, this value should be lowered so that the motor does not oscillate more than a slight overshoot.

WS.TSTANDSTILL can be accessed from the terminal screen. This value determines how long WS.IMAX is applied to the motor. If it is observed that current is not applied for a long enough duration for the motor to come to a complete rest while Mode 1 is executing, then WS.TSTANDSTILL should be increased so that the motor starts motion and comes to a complete rest at a magnetic pole position.

8.3 Feedback 2

The Feedback 2 screen helps configure how you will be using either connector X9 or X7. The screen uses the term feedback in the labels, but you can also think of this as the "signal" source depending on how you use these connection points.



The **Feedback source** box allows you to choose from three feedback sources:

Source 0 – This simple indicates you are not using either connector as a feedback source.

Source 1 – This is tied to the use of connector X9 (consider for emulated encoder connection).

Source 2 – This is tied to the use of connector X7 (consider for the high speed opto inputs on the I/O Connector).

Feedback Mode offers three selections as well, depending how you are using the connection source above.

Mode 0 – The input is configured as A/B signals.

Mode 1 – The input is configured as Step and Direction signals.

Mode 2 – The input is configured as Up-down signals.

The **Resolution** box sets the resolution of the device you have indicated as your input of signal source.

Encoder Emulation Output

The drive offers the flexibility to use connector X9 for an encoder emulation output. This output can be set up using the Encoder Emulation (pg 77) view.

The **Emulation Mode** box offers three settings for Encoder Emulation mode:

Mode 0 – The connector is free to use as an input.

Mode 1 – The X9 connector is now set up as an encoder emulation with a once per rev index pulse (resolution is set in the next parameter (Emulation Resolution).

Mode 2 – The X9 connector is now set up as an absolute index pulse.

Emulation Resolution sets the resolution desired for the output of the emulated encoder.

The **Index or Absolute offset** is dependent on which emulation mode is selected, allowing you to set an offset for either the index pulse or absolute pulse.

A check box allows you to consider the motor direction based on the encoder output.

8.3.1 Encoder Emulation

8.3.1.1 Overview

The encoder emulation (EEO) connector (X9) can be used as either an output or an input. As an output, you can use this connector for position feedback to an analog controller and for configuring the drive as the master in a master/slave system. You can also use X9 connector as an input for controlling the drive through an A/B, pulse and direction, or up/down command. Common applications for this input include using the drive with a stepper controller and configuring the drive as the slave in a master/slave system.

8.3.1.2 Using Encoder Emulation

You can configure the X9 connector from the **Encoder Emulation** screen by setting the connector function, resolution, and (where applicable) input position. The **Electronic Gearing** screen also has provisions for setting the function of the X9 connector.

8.3.1.3 Function Settings

Function settings for the X9 connector are assigned through the DRV.EMUEMODE (pg 514) parameter.

Emulation Mode

0-Input (No EEO Output). While in Emulation Mode, the EEO connector (X9) is configured as an input. This is the recommended setting and should be used in coordination with FB2.MODE (pg 612) to select the type of inputs the secondary feedback will accept. See Feedback 2 (FB2 Parameters (pg 610)) for definitions for the EEO Connector (X9). Note that this mode is also deprecated and will behave as it did with firmware versions prior to M_01-03-00-000.

In earlier firmware versions, this mode indicates that the EEO connector is not operative.

8.3.1.4 Output Modes 1 and 2

The drive generates ouput pulses based on the motor position. Pulse outputs on the SubD connector X9 are three signals: A, B and index, with 90° phase difference (i.e. in quadrature, hence the alternative term “A quad B” output), with a zero pulse.

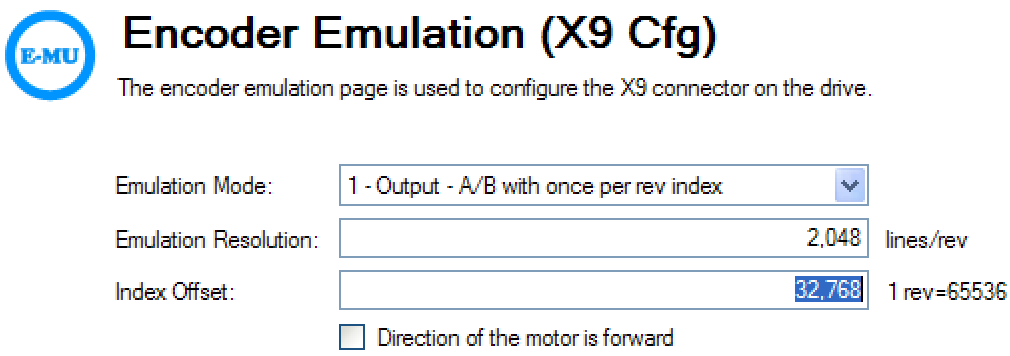
If you are using the AKD as a master, the slave drives use the master's encoder output signals as command input and follows these commands (velocity and direction). The drives operate from an internal supply voltage.

Mode 1– A quad B with once per rev index pulse

Output Mode 1 - A quad B with Once per Rev Index Pulse Connection Diagram

This output mode simulates an encoder signal from the X9 port to another AKD or external controller. EEO resolution (DRV.EMUERES (pg 518)) defines how many counts are outputted for one revolution of the primary feedback.

Index offset (DRV.EMUEZOFFSET (pg 519)) determines the point during the revolution of the primary feedback when the index pulse (X9 pins 7&8) is output through the X9 port. The pulse will occur once every revolution of the primary feedback is at the positive value of the offset. Notice that the offset's resolution is set based on 1rev = 65536, or a 16-bit scale. This is fixed and independent of the Resolution setting of the A and B channel above.



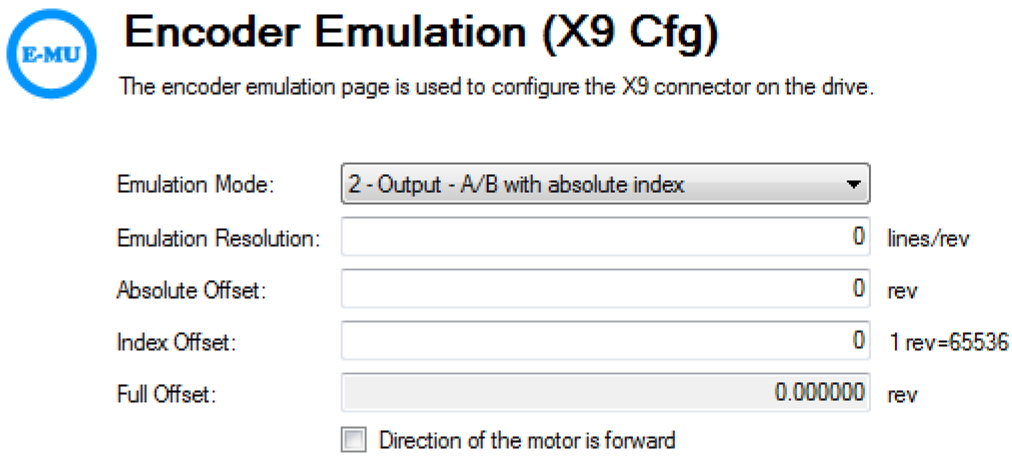
In this case, the Index is offset 180 degrees, or halfway through the revolution of the primary feedback.

Mode 2– A quad B with absolute index pulse

Output Mode 2 - A quad B with Absolute Index Pulse Connection Diagram

When mode 2 is chosen, a box will appear for an absolute index point to be entered (DRV.EMUEMTURN (pg 516)). The absolute index pulse will be output when the motor position reaches the full offset. The sum of the Absolute Offset (the revolutions) and the Index Offset (a fraction of a revolution) make up the "Full Offset". The index offset is fixed as 1rev = 65536, which is a 16-bit scale.

The "Full Offset" is a read-only calculation provided for easier understanding of the placement of the index pulse. In the example below, the pulse will occur at 20.5 positive revolutions of the motor.



8.3.1.5 Input Modes 3, 4, and 5 (deprecated)

The X9 connector is also capable of input modes. These input modes correspond to the signal types described below. The screen also includes provisions for setting the function of the X9 connector for input modes. Because these settings are deprecated, it is recommended to set DRV.EMUEMODE to 0 and use FB2.MODE to select the type of inputs the secondary feedback will accept.

Mode 3-A quad B signals

Input Mode 3 - A quad B Signals Connection Diagram

Input mode 3 allows an A quad B encoder or the encoder emulation output of another drive to be connected and used as a commander encoder, dual loop feedback, gearing, or camming input.

Mode 4-Pulse/direction signals

Input Mode 4 - Pulse/direction Signals Connection Diagram

Input mode 4 allows the drive to be connected to a third-party stepper-motor controller. The number of steps can be adjusted so that the drive can be adapted to match the step-direction signals of any stepper controller.

Mode 5-Up/down signals

Input Mode 5 - Up/down Signals Connection Diagram

The drive can be connected to a third-party controller which delivers up-down signals.

Mode 6- Output - with once per rev index and Input - Step and Direction

This mode allows you to output emulated encoder signals from the X9 (Emulated Encoder) connector, and input a step handwheel signal into the X7 (High Speed Opto IO) connector simultaneously. Mode 6 and 7 are identical, with the exception that the emulated encoder has an incremental Z pulse (Z pulse occurs every revolution) in mode 6, and an absolute Z pulse (Z pulse occurs at one absolute position) in mode 7.

Mode 7 - Output - with absolute Index and Input - Step and Direction

This mode allows you to output emulated encoder signals from the X9 (Emulated Encoder) connector, and input a step handwheel signal into the X7 (High Speed Opto IO) connector simultaneously. Mode 6 and 7 are identical, with the exception that the emulated encoder has an incremental Z pulse (Z pulse occurs every revolution) in mode 6, and an absolute Z pulse (Z pulse occurs at one absolute position) in mode 7.

Resolution

The resolution setting defines how many counts are output for one revolution of the primary feedback (when X9 is configured as an output), or how many counts will be considered a full revolution of the input signal from an external controller (when X9 is configured as an input). The resolution value is post-quadrature; for example, a 1,000 count encoder has a resolution of 4,000 counts.

Note: If the resolution value is set to 0, then the X9 connector will not produce a command.

Related Parameters and Commands

The DRV.EMUEMODE parameter sets the EEO output and input modes. The resolution (before multiplication) is set by the DRV.EMUERES function. The DRV.EMUEZOFFSET parameter adjusts and saves the zero pulse position within one mechanical turn. DRV.EMUEMTURN sets the absolute index point in mode 2 and DRV.HANDWHEEL sets the position at which the index pulse is output in output mode 2.

DRV.EMUEDIR (pg 513)

DRV.EMUEMODE (pg 514)

DRV.EMUEMTURN (pg 516)

DRV.EMUERES (pg 518)

DRV.EMUEZOFFSET (pg 519)

8.4 Multiturn Overflow

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:

```
encoder period = 2^(number of encoder turn bits) (units for one turn of an end system) (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:

```
Machine period ≠ 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.PMTSAVEEN = 1. The supported feedback types include all multiturn encoders: BiSS, EnDAT, Hiperface and Tamagawa (the algorithm requires at least 12 multiturn bits). If FB1.PMTSAVEEN is set to 1 on an invalid feedback type then an error will be reported and FB1.PMTSAVEEN will be set to 0.

Related Topics

FB1.PMTSAVEEN (pg 595) | FB1.PMTBITS (pg 594) | FB1.PSTBITS (pg 599)

8.5 Non-Plug and Play Feedback Devices

To set up a AKD 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 WorkBench, an initial set of parameters can be calculated and downloaded to the drive.

8.5.1 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

8.5.2 Calculations

WorkBench uses the following equations to calculate parameter values.

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*π*(desired bandwidth in Hz)*(motor L line-line in H)

Setting the D and Q components

Velocity Loop

VL.KP = (2*π*75)*(2*Jm/Kt) = 300*PI*Jm/Kt

VL.KI = 5

Slider Tuning

The slider tuning algorithm in WorkBench currently uses the following algorithm.

Input - Motor Data

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

Constants

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

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

Output - Control Loop Gains

VL.KP = 2 * π * BW * Jm * (1+Q) * 0.0001/Kt

VL.KPI = BW*0.08 -1 (minimum of 1)

PL.KP = BW/5

8.6 Foldback

The foldback feature in the AKD 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 WorkBench. The foldback algorithms may reduce the current output to the motor in spite of the current limit settings.

8.6.1 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.

8.6.2 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 WorkBench database.

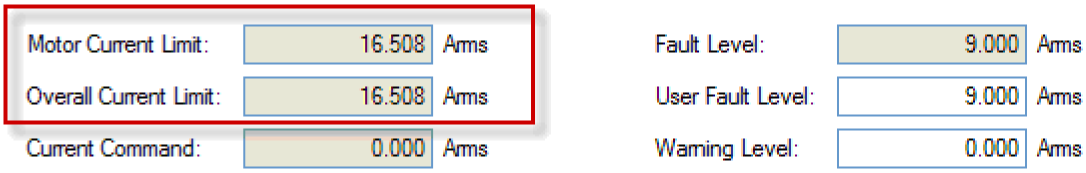
If you are using a custom motor, use the Motor view in WorkBench to set custom values needed for foldback configuration.The parameter entries required for the drive to apply motor foldback protection properly are coil thermal constant (MOTOR.CTFO), peak current of the motor (MOTOR.IPEAK), and continuous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

8.6.3 Setting Fault and Warning Levels

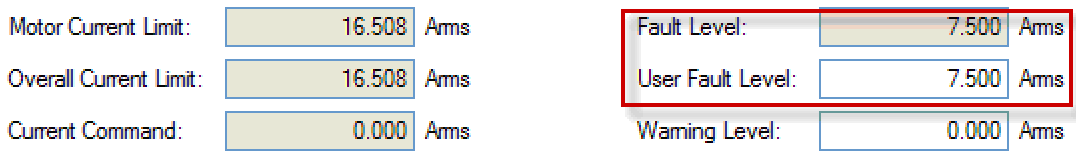
The **Motor Current Limit** and **Overall Current Limit** boxes show status variables that are constantly updated by the foldback algorithm. As current is applied above the drive or motor continuous rating, the capacity for the application of peak current starts to decrease. The motor current limit and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

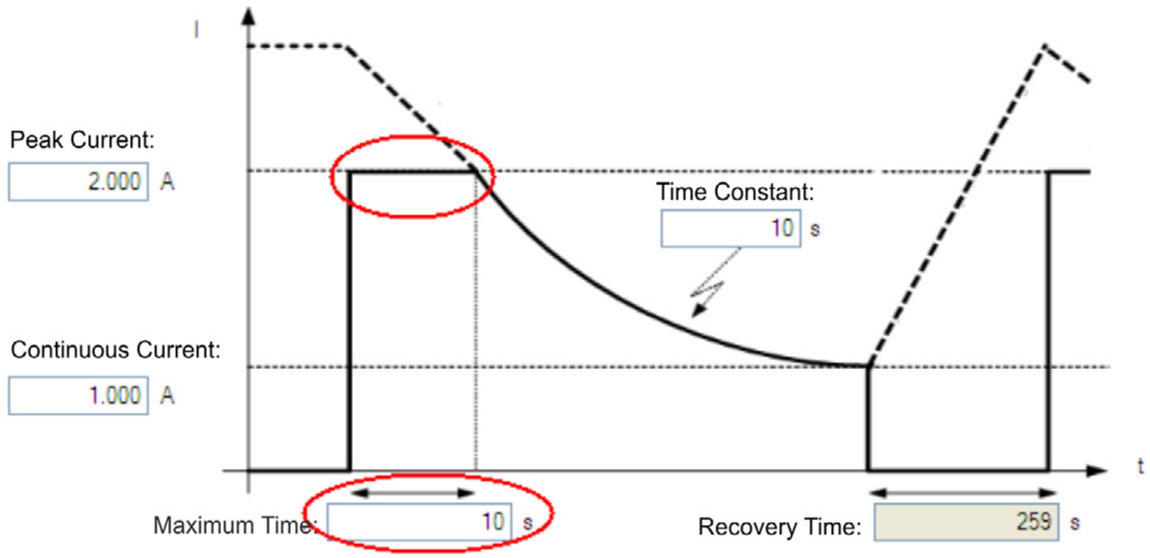


If User Fault Level is set above Fault Level, the User Fault Level will be ignored. The User Fault Level is used to increase the time the drive is operated in foldback mode without faulting. For instance if the default Fault Level is 9.000 Arms and a User Fault Level is set to 7.5 Arms, the Fault Level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



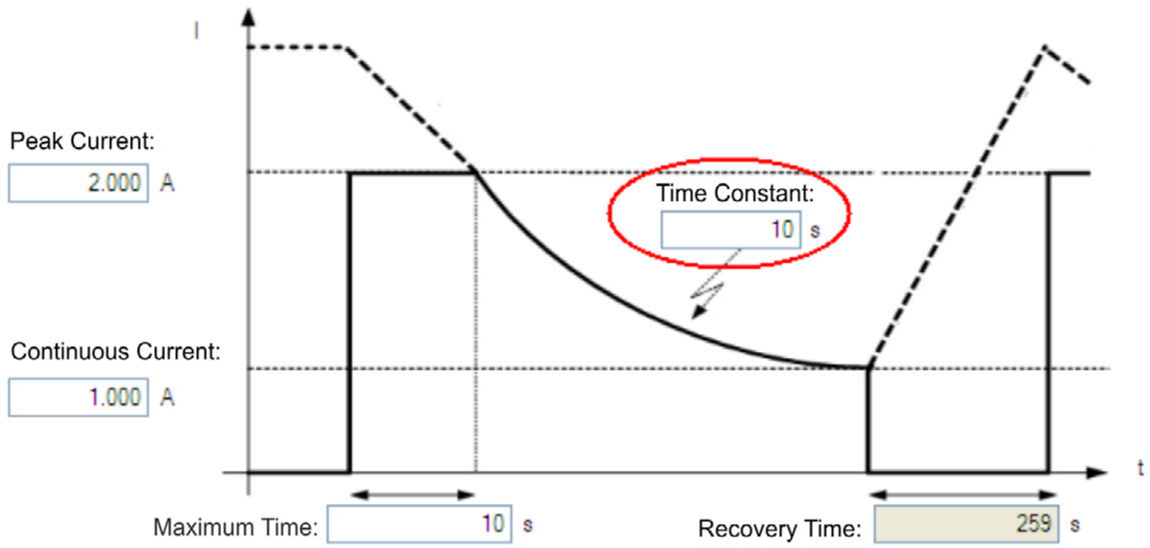
8.6.4 Motor Peak Current Time

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:



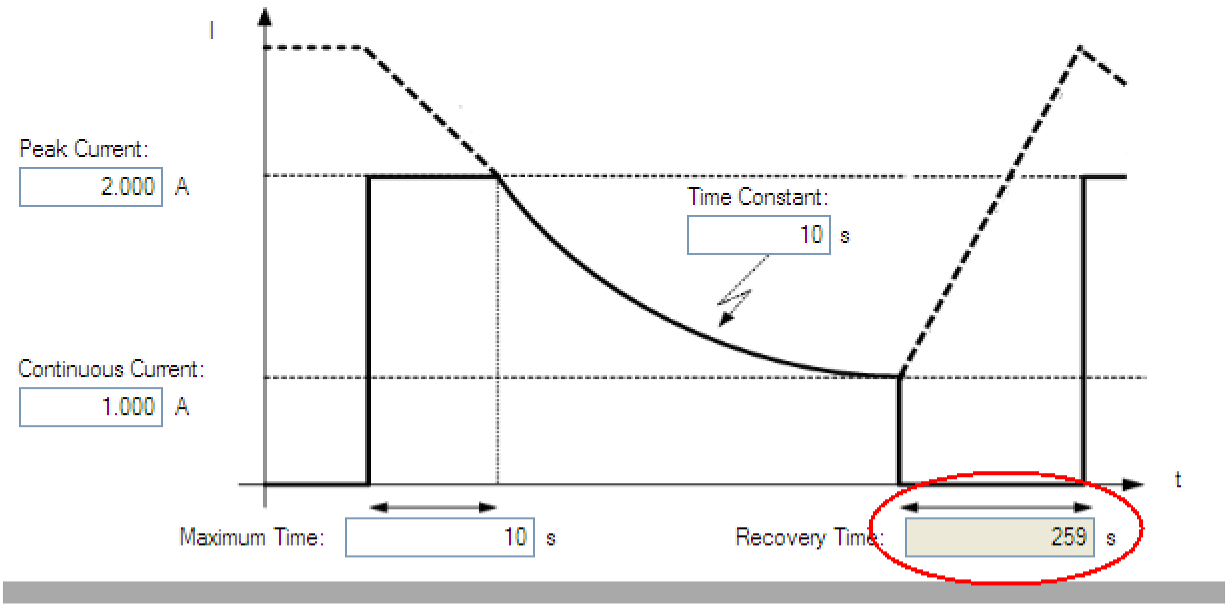
8.6.5 Motor Foldback Ramp

Once the maximum time for motor peak current has elapsed, if the move profile still demands peak current from AKD, 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.



8.6.6 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.



8.6.7 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.

8.7 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 Control	Does this motor have a brake?	MOTOR.BRAKE
Brake Release Delay	The time between the drive being active and the brake being released.	MOTOR.TBRAKERLS
Brake Apply Delay	The time between the brake being applied and the drive not being active.	MOTOR.TBRAKEAPP

8.8 Using Position Capture

8.8.1 Overview

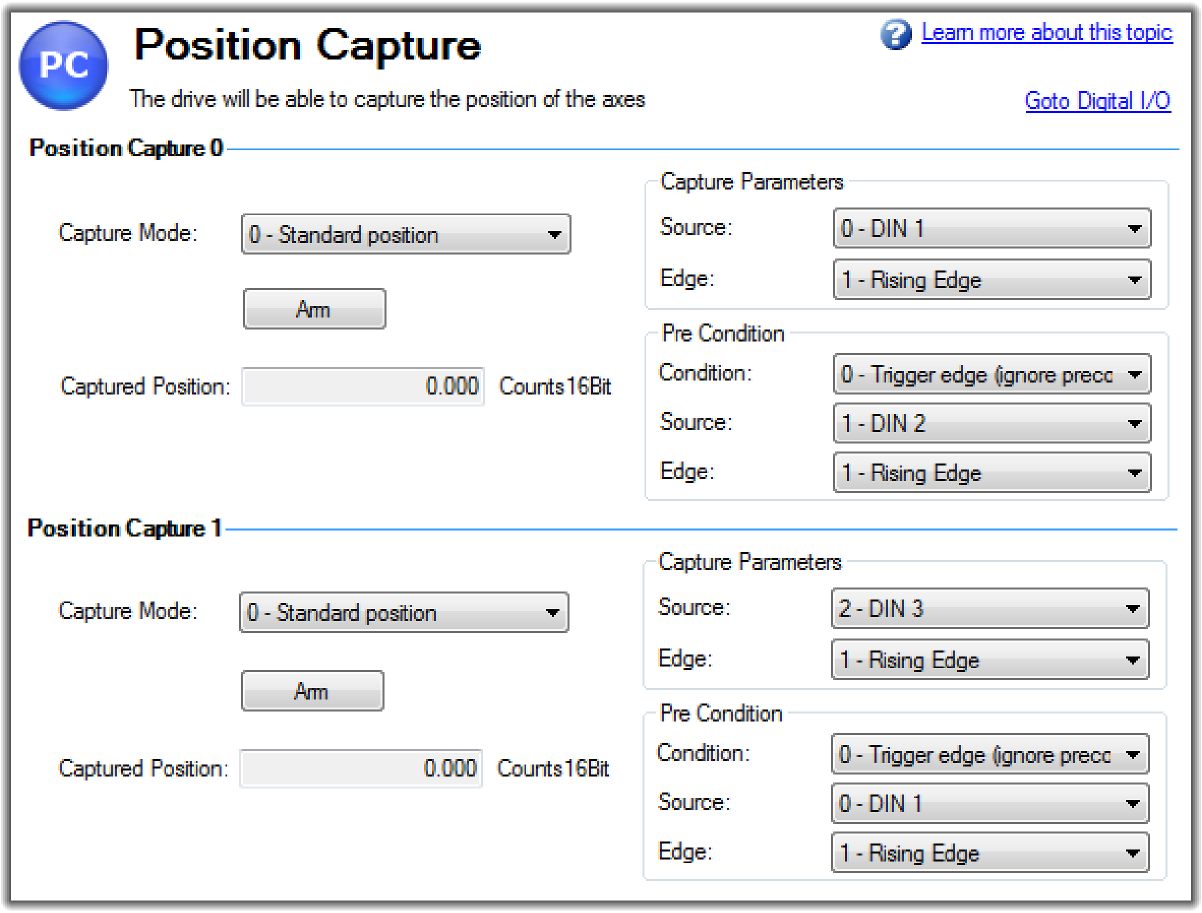
Position capture allows you to precisely determine what the motor position (or drive clock time) was when a specific event triggers. The AKD drive uses two independent captures, which operate similarly. The descriptions provided in this user guide refer to Capture 0, but also apply to Capture 1.

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.

8.8.2 Configuring Position Capture

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



Setting the Capture Source (CAP0.TRIGGER)

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

Capture Source Options:

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 pin through RS485 Input 3 pin, respectively.
10	This option triggers on the primary encoder index.

Setting the Capture Mode (CAP0.MODE)

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

Capture mode options:

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 postion 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 .

Arming and Retrieving the Capture Value (CAP0.EN and CAP0.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).

Setting the Capture Edge (CAP0.EDGE)

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

Capture Edge Options:

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.

Setting the Pre-Condition Event: (CAP0.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.

Setting up a Pre-Condition for complex capture

Setting the Precondition Edge: (Terminal Command: CAP0.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: (Terminal Command: CAP0.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.

8.8.3 Kollmorgen Test Reports

Position Capture test report based on performance testing by Kollmorgen:

```
Capture Accuracy with External Sensor
Drive: AKD-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-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 430)

9 Using AKD 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, it is recommended to set up the axis to immediately apply the brake in the event of the drive disabling for any reason. By setting MOTOR.BRAKEIMM (pg 745) = 1 (default is 0=off) the brake will be applied immediately in the event of the drive disabling.

Drive disabling can occur 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)

NOTE For a digital input initiated controlled stop (A fault that results in controlled stop. See Fault and Warning Messages (pg 250)), or a software disable that is configured for a controlled stop (see DRV.DISMODE (pg 507)), the brake will be applied at the end of the controlled stop.

It is still important to setup the brake delays. This can be done in the brake screen under the settings section in WorkBench. 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 AKD 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 (pg 530) 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).

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 and brake applies.	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 and brake applies.	
	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 and brake applies.	DRV.DISMODE = 3. 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 and brake applies.	
	Fault Dynamic Brake	N/A	N/A
	Fault Power stage disable	None	MOTOR.BRAKEIMM = 1

10 Configuring with Linear Motors

10.1 Connecting a DDL Motor to an AKD Drive

Before Connecting a DDL motor to an AKD 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 cable assembly to the AKD's X10 Feedback Connector.

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

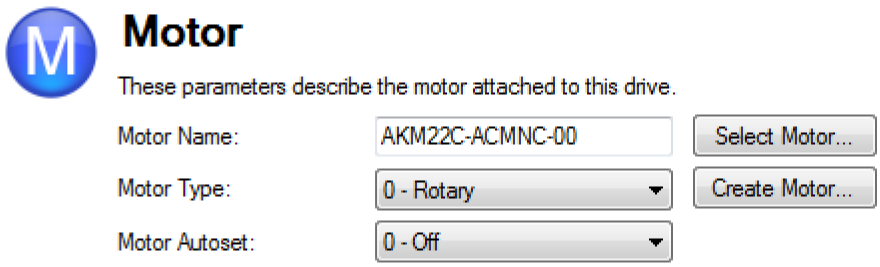
Red -> U

White -> V

Black -> W

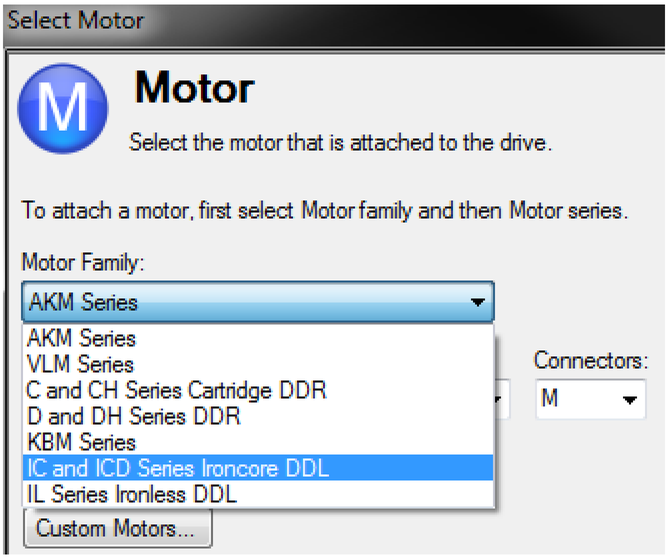
Yellow / Green -> PE

Apply 24 volt logic power to the AKD and launch WorkBench from a computer to interface with the AKD drive. From the main tree, under **Settings**, select **Motor** and click on **Select Motor**.



NOTE If “Select Motor” is grayed out, Motor Autoset may need to be set to “0 –Off” to enable the Select Motor option.

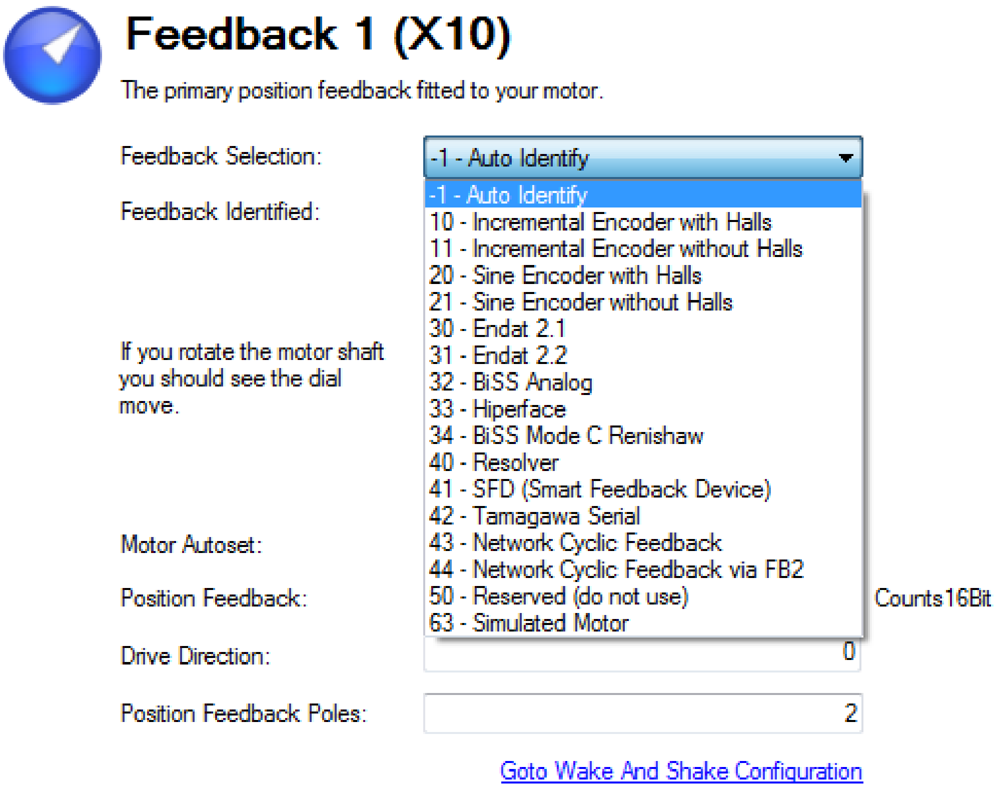
On the Select Motor screen, for Motor Family select either **IC and ICD Series Ironcore DDL** or **IL Series Ironless DDL**. On the Select Motor screen for “Name” select the appropriate motor part number.



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.



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 WorkBench. 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.

11 Selecting Units for Your Application

11.1 Selecting and Saving Units

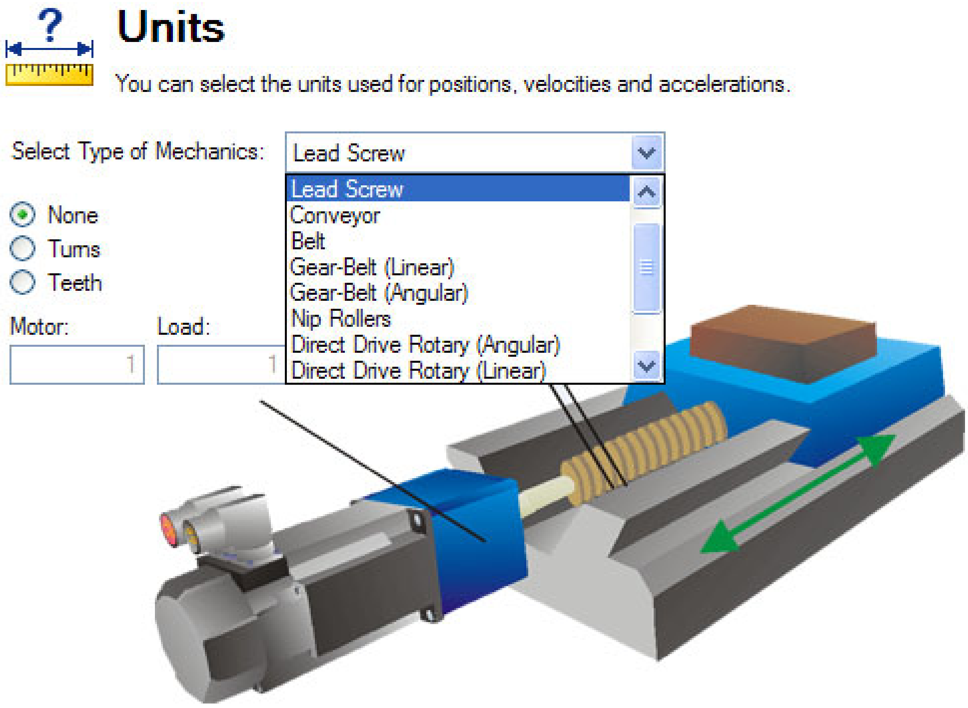
The drive uses three primary measures of motion: position, velocity, and acceleration. You must first choose units of measurement for each of these, and then enter the details of the mechanics in order to scale the chosen units appropriately.

The settings for units are automatically applied. The unit settings in the drive will reflect last settings made in the units screen before exiting. To save settings into the nonvolatile memory of the drive, click the **Save to Drive** button in the toolbar.

11.2 Units Example

1. Select type of mechanics

To set the units in the drive for a particular application, first select the type of mechanics present.



2. Select Position Units

By default, position is measured in counts. Counts are the smallest unit of position that can be represented in the drive. This unit of measure equates to 4,294,967,296 counts/revolution of the motor. You can use the units screen to change this measurement into a meaningful scale relating to the units of the application.

For position units, five selections are available:

0 –Counts (4,294,967,296/rev)

1 –Radians (2*π /rev)

2 –Degree (360/rev)

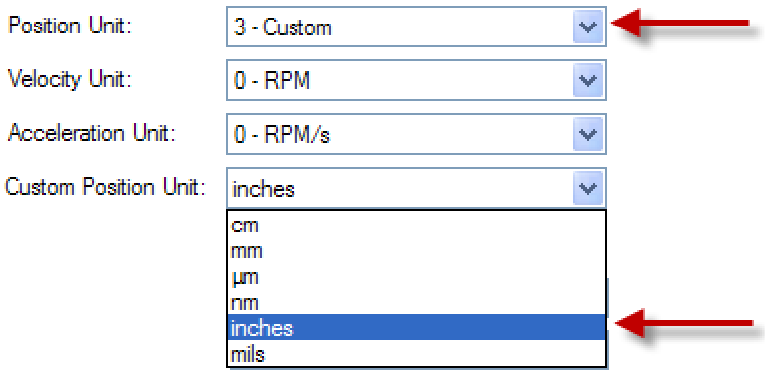
3 –Custom (set by user according to mechanics of machine)

4 –Counts (16-bit) (65,536 /rev)

Custom Units

The drive uses full 32-bit quantization for internal calculations regardless of unit settings. User units settings will not affect performance, resolution, or accuracy of the servo system.

Choose “3-Custom”, and then select your desired position units, for example, millimeters.



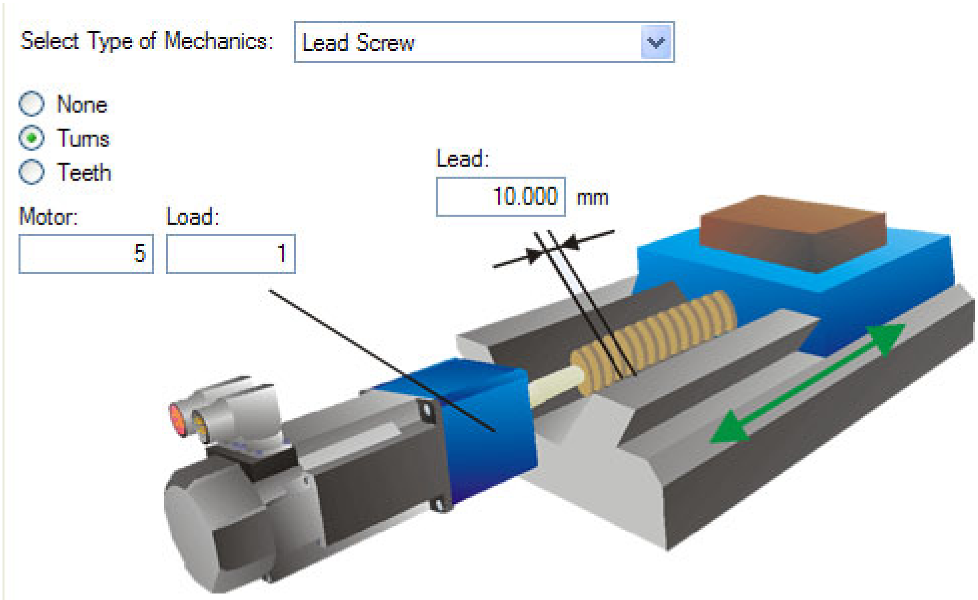
3. Select Velocity Units

For velocity, select **Custom/s** to set the measurement to mm/sec.

4. Select Acceleration Units

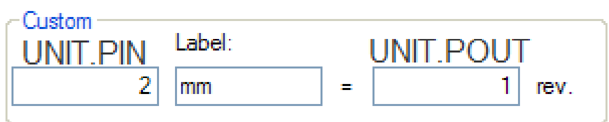
For acceleration, select **Custom/s^2** to set the measurement to mm/sec^2.

Once the chosen units of measurement are established, the details of the mechanics need to be entered to establish the scaling of the chosen units.



In this example, a lead screw with a 10 mm lead and a 5:1 gearbox has been selected. A 10 mm lead means the load will traverse 10 mm for every turn of the screw. A scale factor is applied based on the values entered for the chosen mechanics. This scaling is done through UNIT.PIN and UNIT.POUT parameters in the AKD, which are automatically adjusted when the mechanics values are entered.

UNIT.PIN and UNIT.POUT can be directly entered using the **Motor Only** selection from the **Select Type of Mechanics** box. In this example, scaling is adjusted as follows:



UNIT.PIN is calculated as follows:

10 mm/screw turn * 1 screw turn/5 motor turns = 2 mm/motor turn

Related Parameters

UNIT Parameters (pg 901)

DRV.NVSAVE (pg 547)
MOTOR.TYPE (pg 771)

12 Configuring General Drive Settings

12.1 Digital Inputs and Outputs

12.1.1 Overview

The drive has programmable digital inputs and outputs that you can use to initiate motion, control auxiliary devices, or trigger other actions. The inputs and outputs should be wired according to the instructions in the drive Installation Manual. Note that the input on pin 4 of the X8 connector is dedicated as an enable input.

12.1.2 Using Digital I/O

Once wired correctly, digital inputs and outputs can be used for a variety of functions such as to trigger auxiliary devices, initiate homing moves or other motion tasks, or set travel limits. This section describes the specific functionality of the programmable I/O.

I/O Tip: When using I/O devices, you must carefully consider the type of device you use for switches. An unsuitable switch can cause switch bounce, which in turn can cause erroneous triggers to occur. For example a low cost xx switch, as it is toggled, will bounce a few times before it turns on or off. A device that is monitoring these inputs frequently may interpret the bounce as multiple triggers of that I/O. The drive has the ability to reduce this type of error using some debounce techniques to ignore sudden state changes caused by bounces.

12.1.3 Digital Inputs

Digital inputs can be set in different modes based on the desired function. These functions are outlined below.

Mode 0: Off

This mode is the non-use state and is the default setting for the drive. This mode is valid for all opmodes and command source combinations.

Mode 1: Fault Reset

When an input configured with this mode becomes active, the drive will try to clear all active faults. This mode is edge triggered, so the action occurs only once. If the condition that triggered the fault is still present, the fault condition will remain. See Fault and Warning Messages (pg 250) for details regarding the behavior of individual faults.

This mode is valid for all opmodes and command source combinations.

Mode 2: Start Motion Task

This mode is used to start motion task number x, where x = the value of the associated input parameter. This input will trigger a motion task number as defined in the extra parameter field for this input.

This mode is valid for opmode 2 (position) and command source 0 (service) only.

Once started, the motion task will run until completed. Changes on the input that started the task will be ignored until the motion task is complete. If multiple inputs are configured to start a task, all of these inputs will be gnored until the task is complete. If a motion task is already active in the drive, changes on this input will be ignored.

Example:

-->`DIN1.MODE 2` - sets the input mode to be Start Motion Task

-->`DIN1.PARAM 1` - sets the Motion Task start to 1.

-->`MT.LIST` - confirms that Motion Task 1 exists.

-->`10.000 [counts] 1000.000 [rpm] 0 1001.358 [rpm/s] 1001.358 [rpm/s] 0 0 0 [ms]`

<Create a rising edge of the input>

<Motion Task 1 executed>

Mode 3: Motion Task Select Bit

This mode is used to select the motion tasks that are stored in the drive (numbers 1 to 127) or the reference traverse/homing (0). The motion task number is presented externally at the digital inputs. The motion task set by this mode will be executed when digital input assigned to mode 4 (motion task start selected) gets a rising edge.

This mode is valid for opmode 2 (position) and command source 0 (service) only.

Example

Assume:

DIN1.MODE = DIN2.MODE=DIN3.MODE =3

The state of input 1 and 3 is 1.

The state of input 2 is 0.

Motion task 5 ($5 = 2^0+2^2$) will be executed.

Mode 4: Motion Task Start Selected

This mode is used to start the motion task that is stored in the drive by giving the motion task number. This input uses a secondary variable for the motion task number to be started with the Input trigger. The secondary variable is set by mode 3 (Motion task select bit).

Motion task number “0” initiates homing/reference traverse. A rising edge starts the motion task. A falling edge has no effect.

This mode is valid for opmode 2 (position) and command source 0 (service) only.

Mode 5: Start Home

This mode is used to start the homing motion task on the rising edge. The falling edge has no effect on this input mode of operation.

This mode is valid for opmode 2 (position) and command source 0 (service) only.

Mode 6: Start Jog

This mode is used to start a jog move. This input mode utilizes a secondary variable for the jog’s velocity. The jog will start upon a rising edge. A falling edge stops the jog.

This mode is valid for opmodes 1 (velocity) and 2 (position) and command source 0 (service).

Mode 7: Reserved

Mode 8: Zero Latch

This mode is used to define the current drive position as the zero pulse for the drive EEO and sets the incremental encoder zero pulse offset. The current position, depending on the incremental encoder resolution that is set, is calculated at the rising edge and stored as an offset. An automatic save is then generated. This function is used to perform an automatic setting of the zero pulse in one turn of the motor.

This mode is valid for all opmodes and command source combinations.

Mode 9: Command Buffer

This mode is used to execute four different sets of command buffers. Each set contains two buffers: low and high, for a total of eight buffers. DINx.PARAM for this mode can be 1 to 4, and determines which set of buffers to use.

To set the high and low values of the eight buffers from the terminal screen, use the commands DIN.HCMDx and DIN.LCMDx (1<=x<=4). Use ";" to separate the two buffer commands. Each buffer contains up to 128 characters.

Example

-->`DIN1.MODE 9` (sets command buffer mode to digital input 1)

-->`DIN1.PARAM 1` (sets the first set of buffers to digital input 1)

-->`DIN.HCMD1 DRV.OPMOE 1;` (sets high command buffer)

-->`DIN.LCMD1 DRV.OPMOE 0;` (sets low command buffer)

Under this configuration, a rising edge in digital input 1 will set DRV.OPMODE to 1 and a falling edge will set DRV.OPMODE to 0.

You can also set the command buffers from the **Digital I/O** view in WorkBench; see Command Buffer (pg 112)

This mode is valid for all opmodes and command source combinations.

Mode 10: Control Fault Relay

This mode is used to create an external fault.

Input state is 0 – drive regular behavior

Input state is 1 – “Fault 245 – external fault” is issued.

This mode is valid for all opmodes and command source combinations.

Mode 11: Home reference

This mode is used to receive a physical home reference switch located on the machine to use for the different Home Types.

This mode is valid for opmode 2 (position) and command source 0 (service) only.

Mode 12: Reserved

Mode 13: Controlled Stop

This mode is used to stop the motor using the deceleration variable ramp. If zero velocity is reached, the power stage is then disabled. Also see controlled stop (CS) parameters and commands and Controlled Stop.

This mode is valid for all opmodes and command sources.

Mode 14: Reserved

Mode 15: Quick Stop

This mode is used to stop the motor. It is equivalent to issuing a DRV.STOP command.

This mode is valid for all opmodes and command sources 0 (service) and 2 (electronic gearing).

Mode 16: Activate Electronic Gearing

This mode starts/activates an electronic gearing procedure upon a rising edge.

This mode is valid for opmode 2 (position) and command source 2 (electronic gearing).

Mode 17: Activate Electronic Gear Position Shift

This mode is used to add a position shift to the gearing upon a rising edge. The distance of the position shift is set by the secondary variable. The secondary variable is set by DINx.PARAM. The parameter is in position units and is used to incorporate a phase shift while operating in electronic gearing mode.

This mode is valid for opmode 2 (position) and command source 2 (electronic gearing).

Example

An input is set to add a 180 degree "phase shift" when triggered. As the drive is following the electronic gearing input, the input is triggered and the motor will follow the drive acceleration and deceleration rates to shift 180 degrees while maintaining the gearing synchronization.

Mode 18: Positive Limit Switch

This mode will cause the input to operate as the positive limit switch. If the positive limit switch input is triggered (goes low), the positive direction motion will then be stopped.

This mode is valid for all opmodes and command source combinations.

⚠CAUTION	When setting up the hardware limit switches, you must be certain that the switch remains in the triggered state until you move off of the switch. A very low deceleration rate combined with a high approach velocity may overshoot the switch. This action will cause the position limit warning to be canceled. The warning is not latched, therefore if the switch is overshot, additional movement in the same direction (if commanded) will be possible. This movement can cause machine damage.

Mode 19: Negative Limit Switch

This mode will cause the input to operate as the negative limit switch. If the negative limit switch input is triggered (goes low), the negative direction motion will then be stopped.

This mode is valid for all opmodes and command source combinations.

CAUTION When setting up the hardware limit switches, you must be certain that the switch remains in the triggered state until you move off of the switch. A very low deceleration rate combined with a high approach velocity may overshoot the switch. This action cancels the position limit warning. The warning is not latched, therefore if the switch is overshot, additional movement in the same direction (if commanded) will be possible. This movement can cause machine damage.

Mode 20: Brake Release

This mode is used to apply or release the brake when the drive is not active.

Input = 0: the drive controls the brake (regular drive behavior)

Input = 1: the user controls the brake (apply or release using commands)

This mode is valid for all opmodes and command source combinations.

Mode 21: Current Limit

This mode is used to limit the drive current. The current limit is set by a secondary variable; use DINx.PARAM to set the secondary variable.

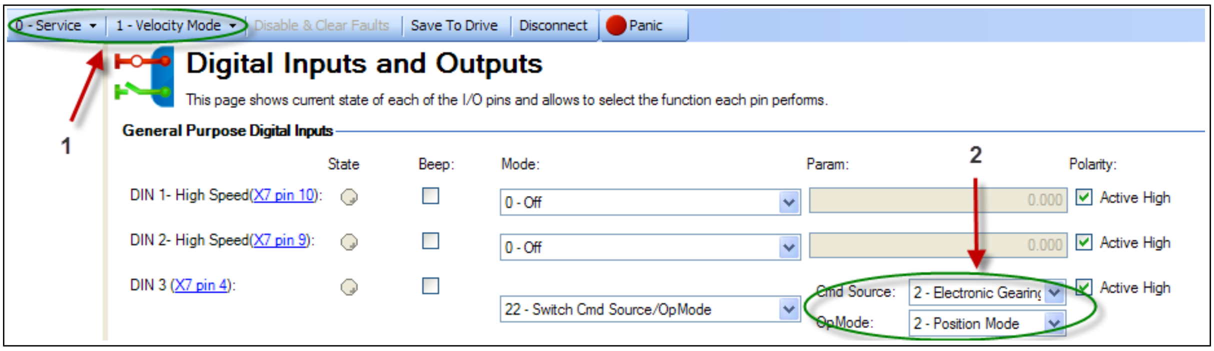
This mode is valid for all opmodes and command source combinations.

Mode 22: Switch Command Source and Opmode

This mode is used to switch between the present command source/opmode and another command source/opmode setting predetermined by the user upon the level change of a digital input. This mode is valid for all opmodes and command source combinations. Arrow 1 in the screen shot below indicates the present command source/opmode that the drive is set for. This is the mode the drive is in when the digital input is not high. This low state is determined by the original settings for DRV.CMDSOURCE and DRV.OPMODE.

The drive will switch into the command source/opmode setting shown in arrow 2 when the digital input level changes to high. This setting is stored by DINx.PARAM and is edited with the drop-down boxes at arrow 2.

NOTE When the digital input is switched high, DRV.CMDSOURCE and DRV.OPMODE will take the values defined by DINx.PARAM. Do not perform a “drive save” in this state, or the low state and high state settings will become the same.



DINX.PARAM	Command Source	Opmode
0	0-service	0-torque
1	0-service	1-velocity
2	0-service	2-position

DINX.PARAM	Command Source	Opmode
10	1-fieldbus	0-torque
11	1-fieldbus	1-velocity
12	1-fieldbus	2-position
N/A	2-electronic gearing	0-torque
N/A	2-electronic gearing	1-velocity
22	2-electronic gearing	2-position
30	3-analog	0-torque
31	3-analog	1-velocity
32	3-analog	2-position

If more than one digital input is configured to this mode, and one of them is active, then the command source/opmode combination configured for that input will be active. If additional inputs become active, the command source/opmode combination configured for the lowest numbered will be the active one.

Example

Assume:

Input 1 is configured for electronic gearing/position.

Input 2 is configured for service/velocity.

Input 3 is configured for fieldbus/position.

The system is in service/torque.

Result:

With no inputs active, the system remains in service/torque.

If input 3 goes active first, then the system will go to fieldbus/position

If input 2 goes active first, then the system will go to service/velocity

If input 1 then goes active, then the lowest active input is now 1 so the system will go to electronic gearing/position

If input 2 then goes active, then the lowest active input is still 1 so there is no change.

If input 3 then goes inactive, then the lowest active input is still 1 so there is no change,

If input 1 then goes inactive, then the lowest active input is 2 so the system will go to service/velocity

If input 2 then goes inactive, then there are no active inputs and the system returns to service/torque.

Mode 23: Change algebraic sign of the measured analog input voltage

This mode can either change the algebraic sign of the measured analog input voltage, or zero the value using a digital input. Since the analog input voltage is used to generate command values in DRV.CMDSOURCE=3 (analog command source), Mode 23 can also be used to change the direction of movement or stop motion using a digital input in DRV.CMDSOURCE=3.

This mode is valid for all opmodes and command source 3 (analog).

The value of DINx.PARAM defines the value of a multiplication factor for the measured analog voltage. The low-byte of the DINx.PARAM value determines the value of this factor, which is multiplied by the measured analog input voltage upon a rising edge on the associated digital input. The high-byte determines the factor upon a falling edge as follows:

DINx.PARAM = 0xFFRR (F=Falling edge; R = Rising edge)

The following values are used in this mode:

Value	Description
0x00	Zero the measured analog voltage.

Value	Description
0x01	Multiply the analog voltage with 1.
0x02	Multiply the analog voltage with -1.
0x03	Zero the measured analog voltage plus trigger in addition a software enable command.
0x04	Multiply the analog voltage with 1 plus trigger in addition a software enable command.
0x05	Multiply the analog voltage with -1 plus trigger in addition a software enable command.
0x06	Zero the measured analog voltage plus trigger in addition a software disable command.
0x07	Multiply the analog voltage with 1 plus trigger in addition a software disable command.
0x08	Multiply the analog voltage with -1 plus trigger in addition a software disable command.

Example 1

DINx.PARAM = 513 = 0x0201

The measured analog input voltage is multiplied with a factor of 1 upon a rising edge on the associated digital input.

The measured analog input voltage is multiplied with a factor of -1 upon a falling edge on the associated digital input.

Example 2

DINx.PARAM = 256 = 0x0100

The measured analog input voltage is multiplied with a factor of 0 upon a rising edge on the associated digital input.

The measured analog input voltage is multiplied with a factor of 1 upon a falling edge on the associated digital input.

Example 3

DINx.PARAM = 1540 = 0x0604

The measured analog input voltage is multiplied with a factor of 1 upon a rising edge on the associated digital input. Additionally, the rising edge on the associated input triggers a software enable command, similar to the DRV.EN command.

The measured analog input voltage is multiplied with a factor of 0 upon a falling edge on the associated digital input. Additionally, the falling edge on the associated input triggers a software disable command, similar to the DRV.DIS command.

Mode 22: Switch Command Source and Opmode

After performing a controlled stop the drive will not be re-enabled when the signal is high. Instead, the drive needs to be re-enabled by the user.

12.1.4 Digital Outputs

Digital outputs can be set in different modes based on the desired function. These functions are outlined below.

NOTE If an output is overloaded (> 100 mA), then the output will turn off (with no indication in WorkBench) and remain off until one of the following occurs:

- The power supply driving the output is removed.
- The output is turned off from the firmware.

NOTE

- The 24V supply to the AKD is power cycled.

If the overload condition still exists, the output will not turn on.

Mode 0-User (Default = 0):The output state is decided by the user or fieldbus. This mode is valid for all opmodes and command source combinations.

Mode 1-Mains Ready: The output mode produces a high signal if the drive DC bus voltage is higher than the under voltage error level and lower than the over voltage error level. This mode is valid for all opmodes and command source combinations.

Mode 2-Software Limit: This output turns on when the software limit positions are reached. This output procudes a high signal if a software limit is reached by traveling in the direction of that software limit. Software limits are set in the **Limits** view. In the **Limits** view, Position 0 is the position limit for negative travel, while Position 1 is the limit for positive travel.

This mode is valid for all opmodes and command source combinations.

Mode 3-Move Complete: When a motion task has completed its move and the trajectory reaches zero and no following tasks are present, the move is considered complete and the output will activate when the actual position is within target_position_area, where target_position_area is as below.

```
target_position_area = motion_task_target_position +/- MT.TPOSWND
```

Mode 3 and Mode 17 (MT in Position) are almost identical. Mode 17 will trigger as soon as the load is in the position window, whereas Mode 3 will wait until the trajectory is complete before monitoring the window. Mode 17 may signal faster because of this, and can also potentially bounce out of the window temporarily.

Mode 4 - Position Error Monitor: This output mode produces a high signal when the absolute value of the position error is less than the parameter entered in the extra parameter field and the drive is enabled.

-DOUx.PARAM < PL.ERR < DOUTx.PARAM

Mode 5-Position Greater than X: When the position is greater than the parameter entered in the extra parameter field, the output will activate.

This mode is valid for all opmodes and command source combinations.

Mode 6-Position Less than X: When the position is less than the parameter entered in the extra parameter field, the output will activate.

This mode is valid for all opmodes and command source combinations.

Mode 7-Warning: This output will activate when the drive experiences a warning, such as positive or negative limit switch input triggered.

This mode is valid for all opmodes and command source combinations.

Mode 8-Enable: If you need an output to indicate that the drive is enabled, use this output mode.

This mode is valid for all opmodes and command source combinations.

Mode 9: Reserved.

Mode 10-Motor Brake: The output mode produces a high signal if a brake is released (this is when the power is applied to the brake and the motor is free to spin). The output mode produces a low if a brake is applied (this is when power is removed from the brake and the brake is set).

This mode is valid for all opmodes and command source combinations.

Mode 11-Drive Faults: The output mode produces a high signal if the drive has a fault.

This mode is valid for all opmodes and command source combinations.

Mode 12-Absolute velocity greater than x: The output mode produces a high signal when the absolute value of the velocity is greater than a variable x. Use DOUTx.PARAM to set x.

This mode is valid for all opmodes and command source combinations.

Mode 13-Absolute velocity less than x: The output mode produces a high signal when the absolute value of the velocity is less than a variable x.

Use DOUTx.PARAM to set x.

This mode is valid for all opmodes and command source combinations.

Mode 14-Homing complete: The output mode produces a high signal when the homing process is completed.

This mode is valid for opmode 2 (position) and command source 0 (service) only.

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.

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.

Mode 16 - Command Buffer Active: The output mode produces a high signal when the commands in a digital input Command Buffer are being executed.

Mode 17 - Reserved

Mode 17 - MT In Position: This output turns on when the position value reaches a window around target position of the active motion task, which doesn’t have any further following motion tasks. The motion task target position window size can be assigned using MT.TPOSWND parameter.

```
target_position_area = motion_task_taget_position +/- MT.TPOSWND
```

Mode 3 and Mode 17 (MT in Position) are almost identical. Mode 17 will trigger as soon as the load is in the position window, whereas Mode 3 will wait until the trajectory is complete before monitoring the window. Mode 17 may signal faster because of this, and can also potentially bounce out of the window temporarily.

Mode 19 – Encoder Z pulse: This is the same signal as the encoder Z signal from encoder emulation (EEO, connect X9). The Z signal from Digital output mode 19 is 24V output and from X9 is RS422 output.

EEO will output the position feedback signal when it is used as an output. For detailed descriptions of EEO see Encoder Emulation (pg 77).

Mode 20 – No Controlled Stop Active: This output mode produces a high signal if no controlled stop is active. When a controlled stop is executed, the signal goes low and stays low until the controlled stop has finished.

Mode 21 – Fault Disabling Power Stage: This output mode produces a low signal when a fault will disable the power stage immediately, or when the hardware enable input is low. The output stays low as long as the fault is not cleared by the user or the hardware enable signal is low.

NOTE DOUTx.STATE always returns 0 in mode 19.

Summary of Opmode and Command Source Dependencies

DINx.MODE	Mode Description	Opmode	Command Source
0	Off	all	all
1	Fault Reset	all	all
2	Start Motion Task	2-Position	0-Service
3	Motion Task Select Bit	2-Position	0-Service
4	Motion Task Start Selected	2-Position	0-Service
5	Start Home	2-Position	0-Service
6	Start Jog	2-Position	0-Service

DINx.MODE	Mode Description	Opmode	Command Source
8	Zero Latch	all	all
9	Command Buffer	all	all
10	Control Fault Relay	all	all
11	Home Reference	2-Position	0-Service
13	Controlled Stop	all	all
15	Quick Stop	all	0-Service
16	Activate Electronic Gear-ing	2-Position	2-Electronic Gearing
17	Electronic Gear Position Shift	2-Position	2-Electronic Gearing
18	Positive Limit Switch	all	all
19	Negative Limit Switch	all	all
20	Brake Release	all	all
21	Current Limitation	all	all
22	Switch CmdSource/Op-mode	all	all
23	Analog In Sign Control	all	3-Analog

DOUTx.MODE	Mode Description	Opmode	Command Source
0	User- (Default=0)	all	all
1	Mains Ready	all	all
2	Software Limit	all	all
3	Move Complete	2-Position	0-Service
4	Position Error Monitor	2-Position	all
5	Position > x	all	all
6	Position < x	all	all
7	Warning	all	all
8	Enable	all	all
10	Motor Brake	all	all
11	Drive Fault	all	all
12	Absolute Velocity > x	all	all
13	Absolute Velocity < x	all	all
14	Homing Complete	2-Position	0-Service
15	Programmable Limit Switch	all	all
16	Command Buffer Active	all	all
17	Mt in position		
19	Encoder Z Pulse	all	all

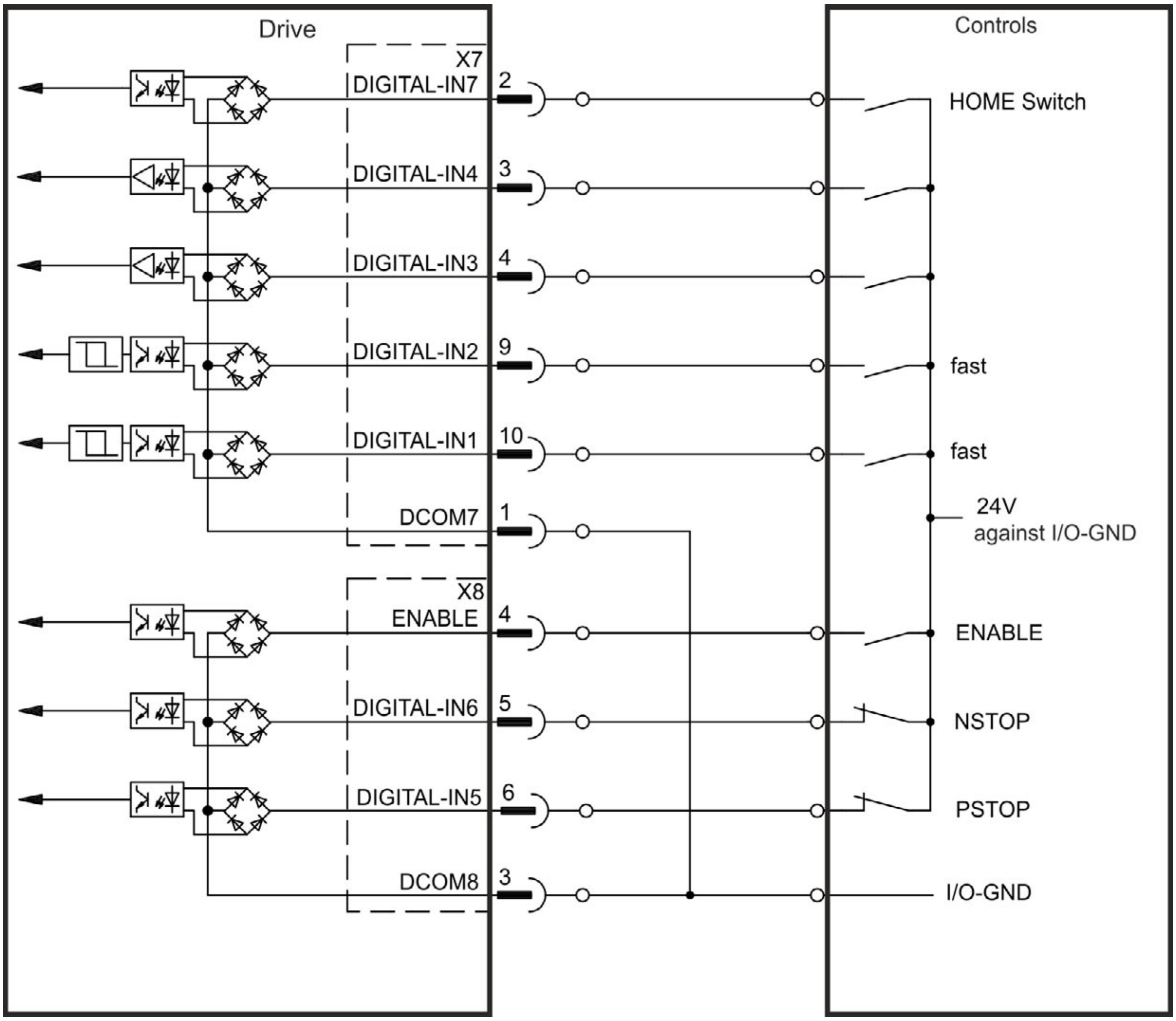
12.1.5 Digital Inputs (X7/X8)

The drive provides 8 digital inputs (➜ p. 295). These can be used to initiate pre-programmed functions that are stored in the drive. A list of these pre-programmed functions is included in the WorkBench. Digital Input 8 is not programmable but is fixed to the ENABLE function.
If an input is programmed, it must be saved to the drive.

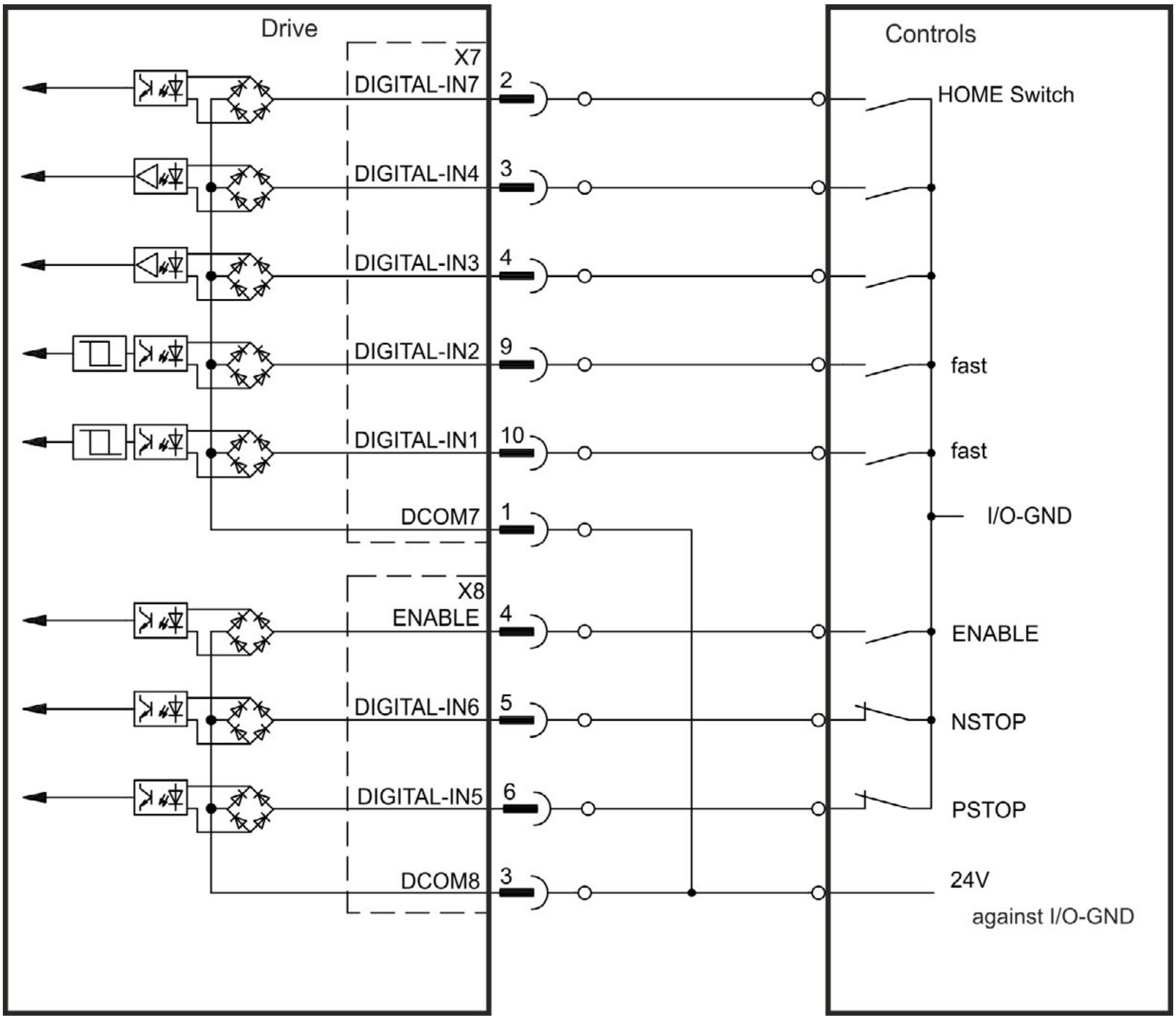
NOTE Depending on the selected function, the inputs are high or low active.

The inputs can be used with switched +24 V (source type) or switched GND (sink type). See diagrams below for typical examples of digital input wiring.

Digital Input Wiring Diagram (Source type connection, example)



Digital Input Wiring Diagram (Sink type connection, example)



12.1.5.1 Digital Inputs 1 and 2

These inputs (X7/9 and X7/10) are particularly fast and are therefore suitable for latch functions, for example. They can also be used as 24 V inputs for electronic gearing .

Technical characteristics

- Floating, reference common line is DCOM7
- Sink or Source type sensors possible
- High: 3.5 to 30 V/2 to 15 mA , Low: -2 to +2 V/<15 mA
- Update rate: Hardware 2 µs

12.1.5.2 Digital Inputs 3 to 7

These inputs are programmable with the setup software. By default,all inputs are not programmed (off).

For more information refer to the setup software.

Technical characteristics

Choose the function you require in WorkBench.

- Floating, reference common line is DCOM7 or DCOM8
- Sink or Source type sensors possible
- High: 3.5 to 30 V/2 to 15 mA , Low: -2 to +2 V/<15 mA
- Update rate: Software 250 µs

12.1.5.3 Digital Input 8 (ENABLE)

Digital Input 8 (terminal X8/4) is set to Enable function.

- Floating, reference common line is DCOM8
- Sink or Source type wiring is possible
- High: 3.5 to 30 V/2 to 15 mA , Low: -2 to +2 V/<15 mA
- Update rate: direct connection to hardware (FPGA)

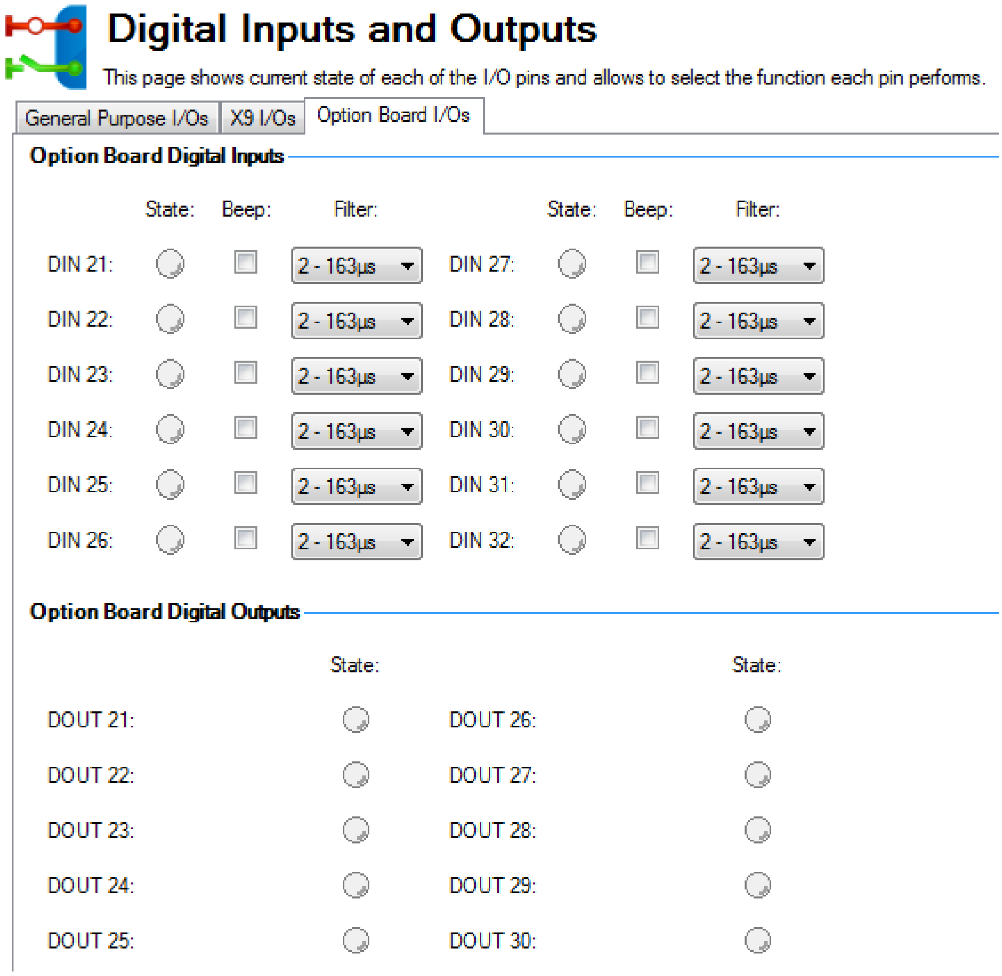
NOTE The Hardware Enable input and the Software Enable signal (via fieldbus or WorkBench) are serial, that means wiring of Hardware Enable is mandatory.

The output stage of the drive is enabled by applying the ENABLE signal (Terminal X8/4, active high). Enable is possible only if input STO has a 24 V signal . In the disabled state (low signal) the connected motor has no torque.

A software enable by means of the setup software is also required (AND link), although this can also be permanently enabled with WorkBench.

12.1.6 Option Board I/Os

If connected to a drive equipped with extended I/Os (X21, X22, X23, X24) select the Option I/Os tab from the Digital I/O view to configure digital input and digital output parameters.



Related Topics

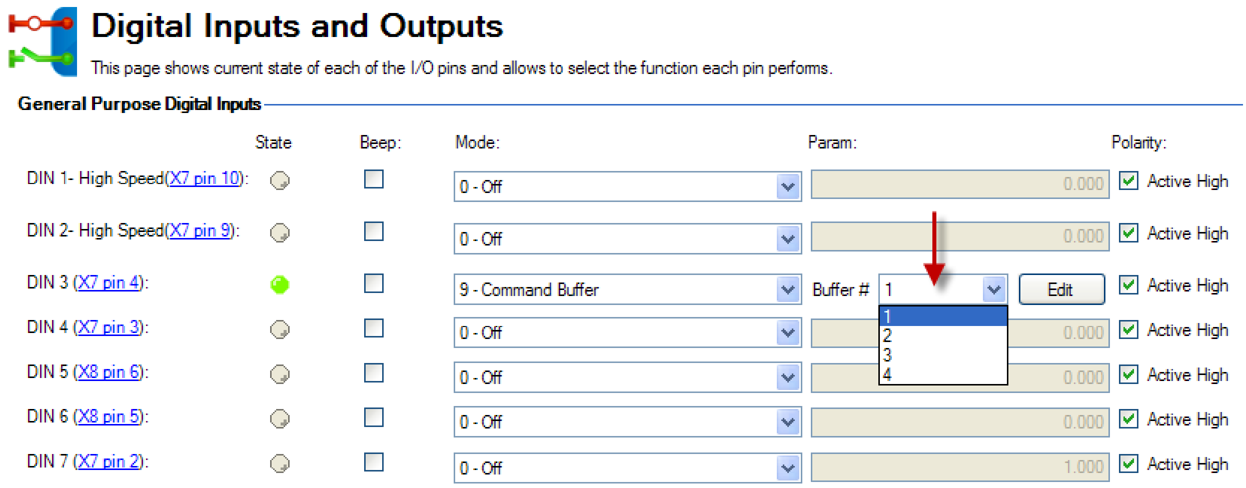
DIN21.FILTER to DIN32.FILTER (pg 467) | DIN21.STATE to DIN32.STATE (pg 468) | DOUT21.STATE to DOUT32.STATE (pg 485) | DOUT21.STATEU to DOUT32.STATEU (pg 486)

12.2 Command Buffer

12.2.1 Overview

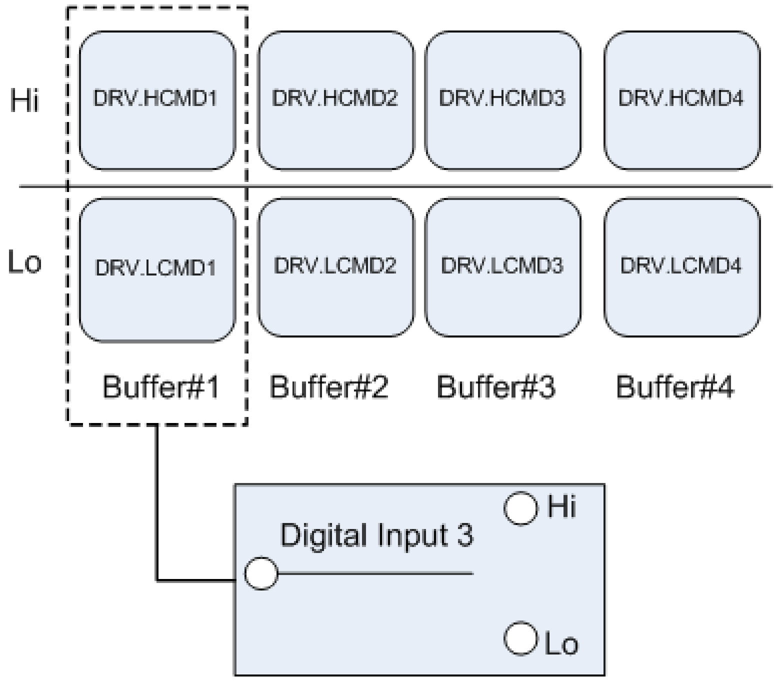
The Command Buffer input mode (Digital Inputs and Outputs (pg 100)) allows you to change values for parameters using a digital input.

The drive has four available buffers. A digital input configured for command buffer mode is linked to one command buffer set. This is determined by the user (see arrow 1). In this case, command buffer 1 is used.



The graphic below explains the architecture of the buffers.

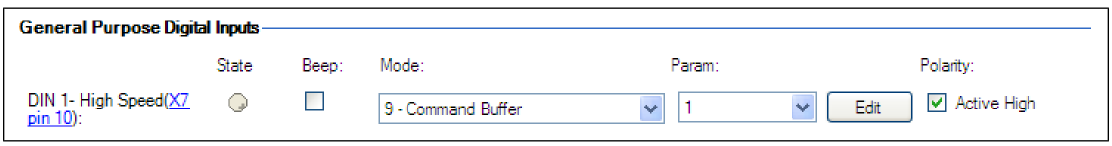
Available buffers in AKD:



12.2.2 Editing the Command Buffers

By default, the buffers are empty. Each side of the buffer can contain 128 characters maximum (parameter and value included). You can use the **Command Buffer Editor** to enter the sequence of commands to the digital input buffer. You can enter the sequence of commands into Low command buffer (DIN.LCMDx) or High command buffer (DIN.HCMDx) and save these settings to the drive.

To edit the command buffers, open the **Digital I/O** view and select **Digital input mode** as **9 – Command Buffer**.



The **Param** box lists the available command buffers. Select desired command buffer number for the DIN. This number sets to the DINx.PARAM keyword. When you mouse over the **Param** box, the Tooltip displays the current content of the High command buffer and Low command buffer in the drive.

To edit the selected command buffer, click **Edit** to display the command buffer editor screen.

The command buffer editor screen has following properties:

Button or Dialog Box	Description
Command Buffer Number	The identification number of the command buffer (1, 2, 3, 4).
High Command Buffer	Adds sequence of commands to the High command buffer parameter. Contents are saved to the keyword DIN.HCMDx. A maximum of 128 characters can be set to drive along with the separator “;”. Commands must be entered line by line and when saving to the drive each commands will be formed into single line separated by “;”.
Low Command Buffer	Adds sequence of commands to the Low command buffer parameter. Contents are saved to the keyword DIN.LCMDx. A maximum of 128 characters can be set to drive along with the separator “;”. Commands must be entered line by line and when saving to the drive each commands will be formed into single line separated by “;”.
Ok	Saves the sequence of commands to the drive.
Close	Closes the screen and returns to Digital I/O view. If contents are not saved to drive before closing the screen, confirmation message “The commands have been modified and not save to drive. Do you want close without saving?” is displayed.

Commands and parameters are entered on separate lines with a space between the parameter and the value.

A semicolon separator is not necessary in the editor, but it is required if the buffers are edited inside the terminal window.

12.2.3 Behavior of the Command Buffer

Digital inputs have either a high or a low state. The contents of the buffer are executed at the rising edge of the state change. The contents of the buffer are also loaded at drive power up according to the starting state of the digital input. When the command buffer is initially configured, the buffer is not executed until the first digital input state change is detected.

Tip: Once you have the buffer configured and tested, put the digital input in the most common state that it will be in at start up. Save the parameters to the drive. This will synchronize the NVRAM with the buffer, so at start up, values will not have to be changed.

12.2.4 Delays for the Buffer

There is a delay command that can be used to delay the execution of a command or parameter change (DRV.CMDDELAY). The value can be from 0 ms to 5000 ms.

The following commands may require a delay before the next command in the buffer can be executed:

DRV.EN (100 ms min)

DRV.DIS (50 ms min)

Example:

```
DRV.EN
DRV.CMDDELAY 100
MT.MOVE
```

The command buffer does not send back warnings when a parameter is invalid or out of range, so make sure the syntax is correct and that the digital input changes during legal drive states for the commands given.

12.3 Analog Input

If the drive command source is set to analog, then the analog input to the drive supplies the current or velocity command to the control loops of the drive.The default analog input screen displays a summary block diagram of the analog input. You can adjust the analog input settings from this view as follows:

Button or Dialog Box	Description	Parameter
Offset	The offset adds a bias to the analog input command. This offset is commonly used to remove any biases that may be present on the analog input signal.	AIN.OFFSET (pg 372)
Input Voltage	The value of the analog input after the offset, deadband, and low pass filters.	AIN.VALUE (pg 375)
Scale	If the opmode is current mode, then this value is the amount of current that will be commanded for each volt on the analog input. If the opmode is velocity mode, then this value is the velocity that will be commanded for each volt on the analog input. **Note:** Scaling in KAS is 80% of that in AKD. In AKD 10 v = 32767 while in KAS 10V = 26126	AIN.ISCALE (pg 370), AIN.VSCALE (pg 376), or AIN.PSCALE (pg 373)
Torque Command	The current or velocity command that is sent to the control loops.	IL.CMD (pg 687)PL.CMD (pg 809)or VL.CMD (pg 931)

Click the **More** button to access a detailed view of the analog input. You can adjust additional analog input settings from this view as follows:

Button or Dialog Box	Description	Parameter
Low Pass Filter	The break point frequency for the low pass filter.	AIN.CUTOFF (pg 365)
Deadband	The threshold for the deadband. This parameter is commonly used to reduce noise while the drive is stationary.	AIN.DEADBAND (pg 366)

Both analog input views provide a link to the encoder emulation output setup; see Encoder Emulation (pg 77)for more details on this feature.

Related Parameters

AIN Parameters (pg 364)

12.4 Analog Output

The drive has one analog output. You can either directly control the voltage output or select a different mode to output different signals. Analog output is configured through the X8 connector; for detailed information on this connector, see Connection Diagrams, Analog Output.

Button or Dialog Box	Description	Parameter
Analog Output Mode	Select which internal signal is output by the analog output.	AOUT.MODE (pg 396)
Analog Output Value (user)	Enter the analog output value (when AOUT.MODE (pg 396) = 0, analog output signal is determined by the user)	AOUT.VALUEU (pg 402)
Analog Output Value	Displays the voltage that this analog output generates.	AOUT.VALUE (pg 401)
Analog Velocity Scale Factor	Scales the analog output (AOUT.VALUE (pg 401)) for AOUT.MODE = 1, 2, or 3.	AOUT.VSCALE (pg 403)
Analog Output Offset	Sets the analog output 2 offset.	AOUT.OFFSET (pg 398)
Low Pass Filter	Enables a software-based low pass filter of the analog output value. 0 Hz is a pass-through or "disable" of this feature.	AOUT.CUTOFF (pg 394)

The analog output modes consist of the following:

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)

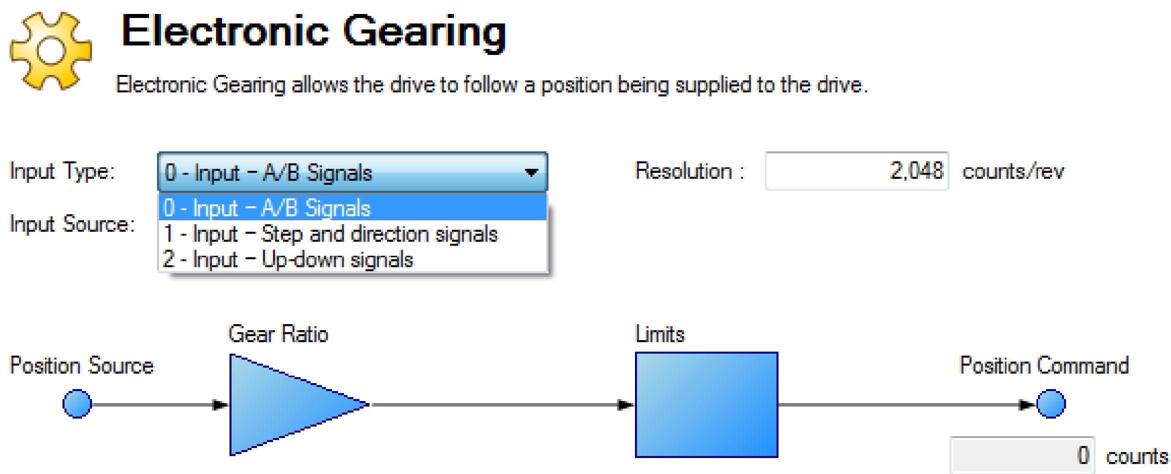
12.5 Electronic Gearing

12.5.1 Overview

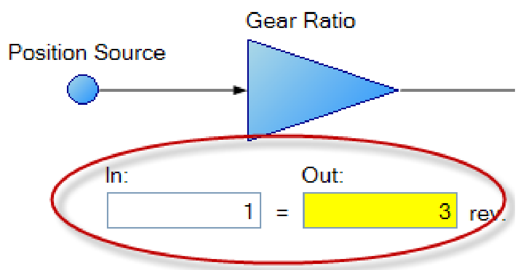
Electronic gearing is the act of sending a digital pulse position command to the AKD drive. The X9 connector is used for controlling the drive through an A/B type signal, pulse and direction (also called step and direction), or up/down command. A common application for electronic gearing is using servos with a stepper controller or daisy chaining multiple AKD drives from one master drive as slave drives.

In order to command an AKD using electronic gearing, the command source (DRV.CMDSOURCE) must be set to **2-Electronic Gearing** and the opmode (DRV.OPMODE) must be set to **2-Position Mode**.

The input modes of the X9 connector are used to set up the AKD for electronic gearing.



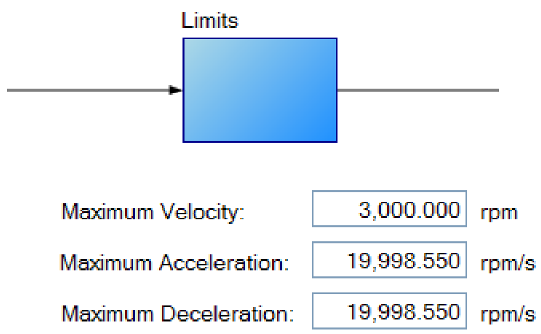
Resolution is the post-quadrature value of the counts/revolution of the input. Additionally, a gear ratio can be applied to affect the output ratio of the motor.



The position command (DRV.HANDWHEEL) reads the EEO value, where 4,294,967,296 is a full revolution of the input, then the value rolls over. Gear ratio does not affect the EEO value. If the output is set to 3 output revs per input rev, there will be 4,294,967,296 counts per 3 revolutions of the motor.

12.5.2 Limits

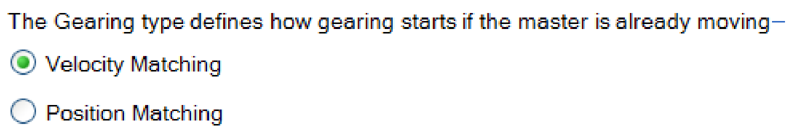
Electronic Gearing has independent limits, as shown below:



These limits (GEAR.ACCMAX, GEAR.DECMAX, GEAR.VELMAX) are applied only during gearing mode and the units are consistent with speed and acceleration of the output motor. All other limits in the drive are active along with gearing limits.

NOTE Since the master determines the trajectory profile of the slave, it is not common to need to set GEAR limits to change the profile from the master. Erratic motion may result since they can clamp the servo command. If you are experiencing problems, first increase these limits to their maximums.

If the master is already moving when entering electronic gearing mode, velocity or position can be matched (GEAR.MODE):



In **Velocity Matching**, the motor will ramp up to the same velocity with no concern over lost steps during the acceleration period.

In **Position Matching**, the motor will match the position command from the switchover point by speeding up to recover the lost steps during the acceleration period.

12.5.3 Determining Maximum Cable Length

When you use an external incremental encoder as an input to X9, you must determine the maximum allowable cable length.

NOTE This information is only applicable when using an external encoder as either a secondary feedback input or a gearing command (DRV.EMUEMODE3). Not applicable for any other X9 mode or when using two AKDs in a master/slave system.

The X9 port has a 5V output used to supply power to an external incremental encoder.

The maximum cable length depends on the current draw of the external encoder and the cable type connecting the X9 port. The following example can be used as a guide to calculate the maximum cable length for your application.

X9 port characteristics:

Nominal Supply Voltage:5 V

Tolerance: 5%

Minimum Supply Voltage: 4.75 V

Maximum current: 0.25 A

Permitted wire gauge: 20-28 AWB (Typical for D9 connector)

Sample Application Hardware:

Example external encoder: Hengstler RI-36H (RS-422 encoder) used with X9 port.

Encoder Nominal Supply Voltage: 5V (+/- 10%)

Minimum Supply Voltage: 4.5 V calculated based on tolerance above

Maximum required encoder supply current: 50 mA

Example cable:

Lapp Li2YCY - 24AWG (0.22 mm^2)

Loop resistance: 0.186 Ohms/m

Sample Calculations:

Maximum Permissible voltage cable drop = 0.25 V

= (Minimum Supply Voltage from AKD) 4.75 V – (Minimum supply voltage of RI-36H encoder) 4.5V

Maximum permissible resistance of cable run to X9 = 5 Ohms

= (Max voltage cable drop) 0.25V ÷ (Maximum encoder current) 0.05A

Maximum permissible cable length for example application = 26.9 m

= (Max cable resistance) 5 Ohms ÷ 0.186 Ohms/m

Related Parameters

GEAR Parameters (pg 639)

DRV.CMDSOURCE (pg 497)

DRV.EMUEMODE (pg 514)

DRV.EMUERES (pg 518)

DRV.HANDWHEEL (pg 525)

DRV.OPMODE (pg 549)

12.6 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 (pg 710)
Negative Peak Current	The maximum negative current allowed.	IL.LIMITN (pg 709)
Velocity Limits		
Positive Speed Limit	The maximum speed allowed in the positive direction.	VL.LIMITP (pg 949)
Negative Speed Limit	The maximum speed allowed in the negative direction.	VL.LIMITN (pg 947)
Over Speed Limit	The threshold velocity for an over speed fault.	VL.THRESH (pg 955)
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 (pg 811)
Position Limit 0	The minimum position the drive can reach before generating a negative software position fault.	SWLS.LIMIT0 (pg 895)
Position Limit 1	The maximum position the drive can reach before generating a positive software position fault.	SWLS.LIMIT1 (pg 896)
Acceleration Limits		

Button or Dialog Box	Description	Parameter
Acceleration	The acceleration ramp used to profile some types of motion.	DRV.ACC (pg 489)
Deceleration	The deceleration ramp used to profile some types of motion.	DRV.DEC (pg 501)
Motor Limits	Motor limits are set through the Motor Foldback Screen (see Foldback (pg 81))	

12.6.1 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 then 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. Position Limit 0 is tied to the clockwise (positive) motor direction. As the motor reaches the position entered, the motor will stop and display a warning n107. Position Limit 1 is tied to the counterclockwise (negative) motor direction. As the motor reaches the position entered, the motor will stop and display a 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.

12.7 Programmable Limit Switch

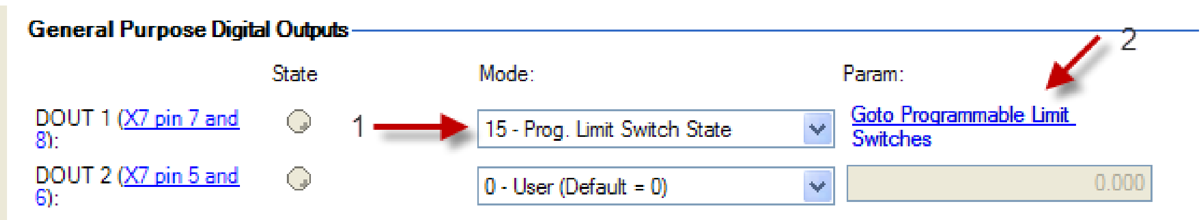
12.7.1 Overview

Programmable Limit Switches (PLSs) are used to turn on and off drive digital outputs based on the drive's position. Multiple positions can be combined to affect the state of an output when PLSs are combined.

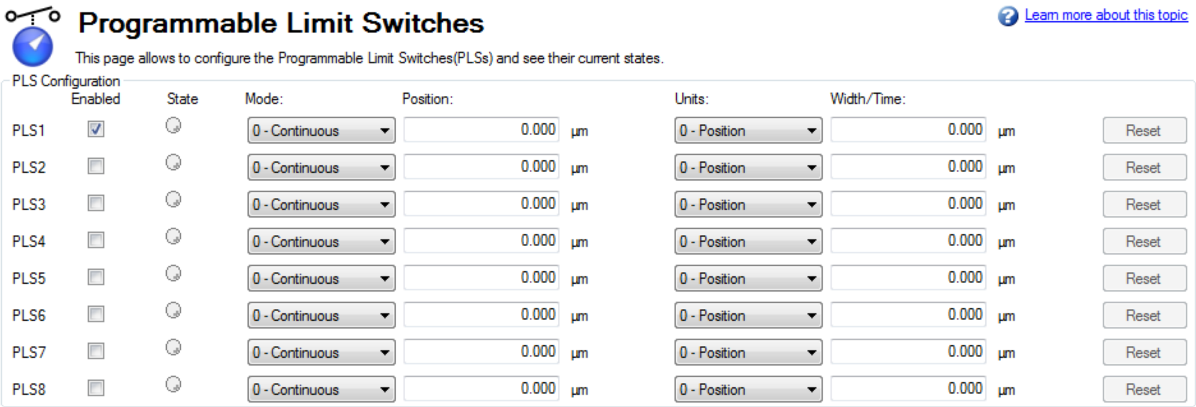
12.7.2 Using Programmable Limit Switches

To use PLSs, you must first configure a digital output as follows:

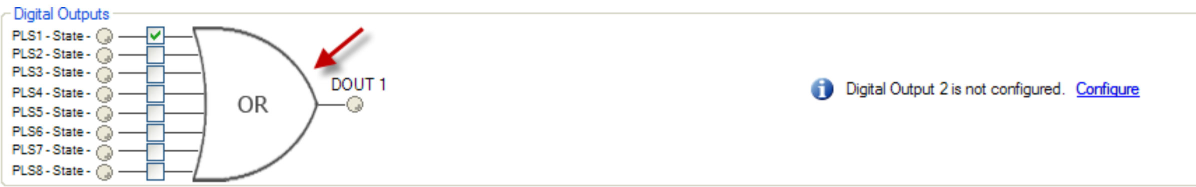
1. Click on the Digital I/O icon in the the tree view.
2. Set the output of your choice to mode fifteen (see 1 below). In this example, digital output 1 is used.
3. Now that the digital output mode is set for PLS, you can click on the **Goto Programmable Limit Switch** link (see 2 below) to open the PLS screen (this screen is also shown in the WorkBench tree view).



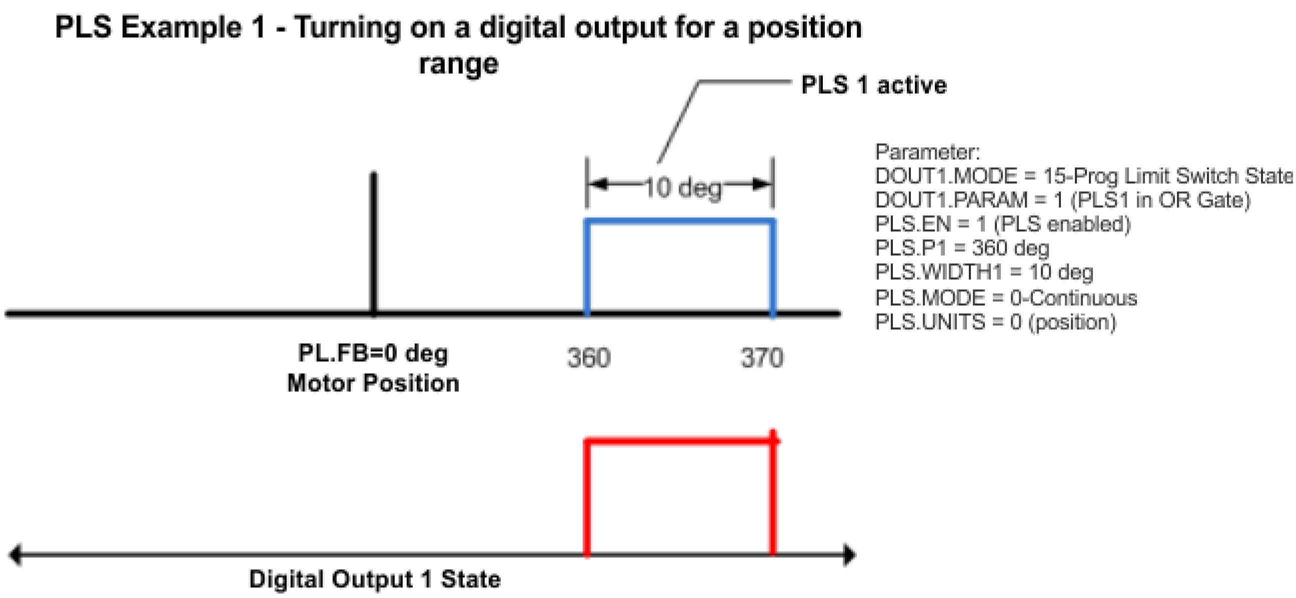
The PLS screen is used to establish the positions for the output(s) to turn on.



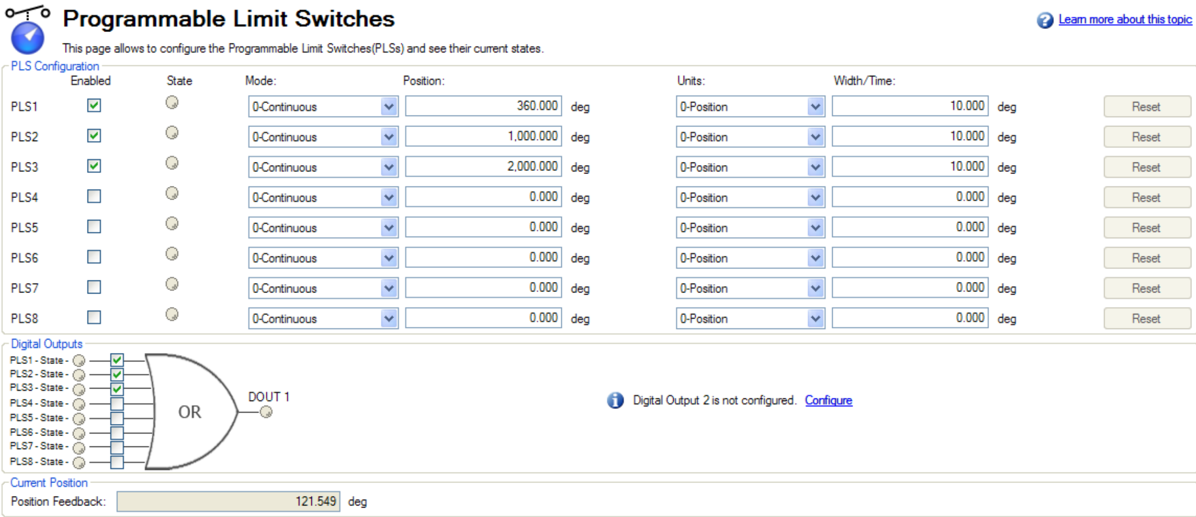
The PLS configuration section of the screen sets the mode and limits of each of the eight PLSs. The PLS is ignored unless it is enabled (see image above). In the screen example, PLS1 is set for continuous operation in position mode. Every time the 360 degree position (PL.FB) is crossed in either direction, the output will turn on for 10 degrees of motor movement.



The final step is to configure the OR gate for the PLSs on which output is triggered. The gate appears for setup in the screen when a digital output is configured in Mode 15 – Prog Limit Switch State. Since only PLS1 is configured, select PLS 1 (see arrow above)

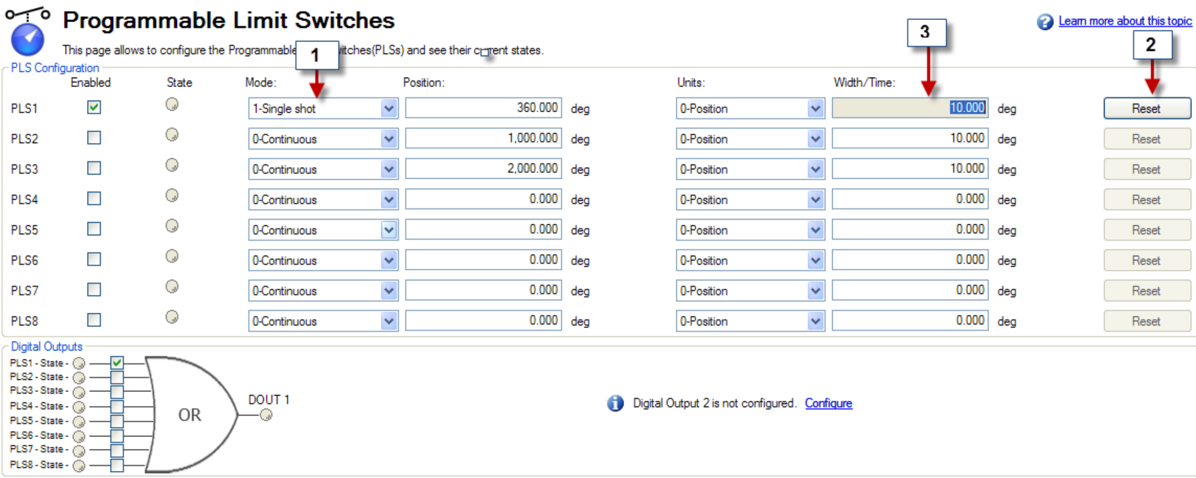


To setup an output with multiple turn-on points, configure and enable more PLS's and include them in the OR Gate.

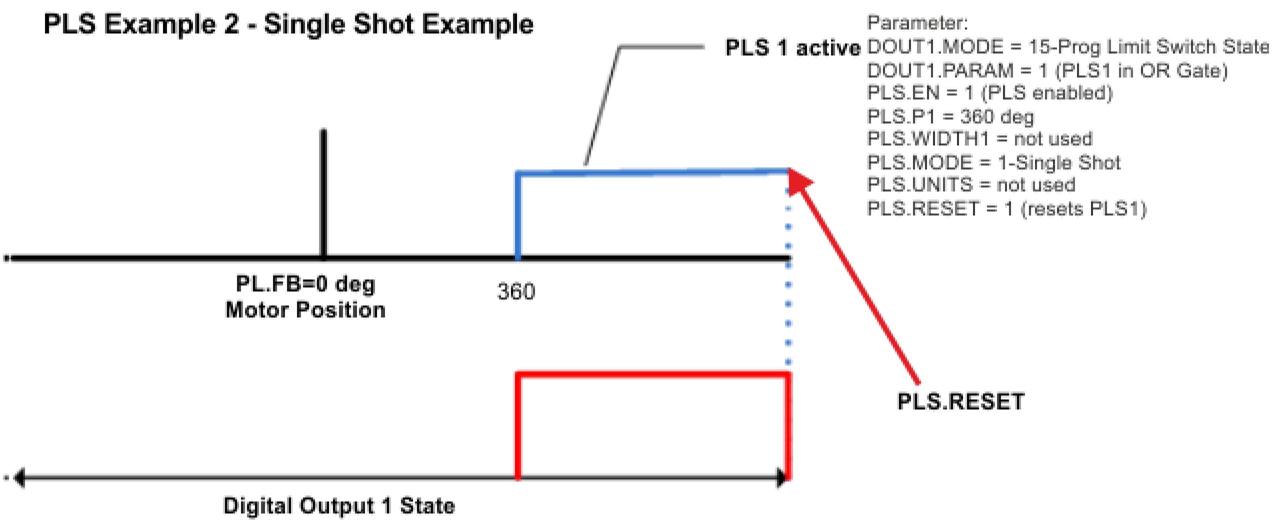


12.7.3 Single Shot Mode

Single shot mode is a special mode of PLS. Single shot mode (see 1 below) turns on the output until it is reset (see 2 below). Normal operation of this mode usually depends on a machine controller to reset the PLS using the fieldbus object for PLS.RESET .



Single Shot Example:



Related Parameters

PLS Parameters (pg 830)
Digital Inputs and Outputs (pg 100)

12.8 Enable/Disable

12.8.1 Enable Modes

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

Hardware Enable Mode

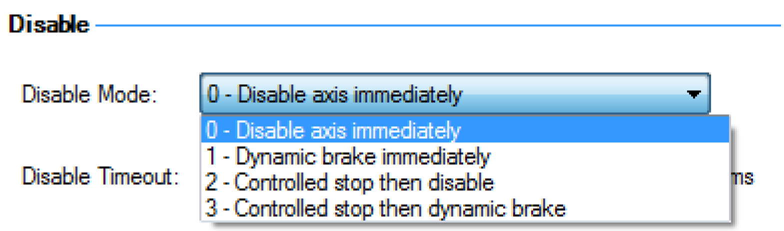
The AKD has two methods hardware enable methods. These methods are controlled by DRV.HWENMODE. Mode 0 allows for the drive to enable and clear faults on the rising edge of the hardware enable input. Mode 1 will NOT clear any faults on the rising edge of the hardware enable input, allowing you to review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

Software Enable Default

In addition, the Software Enable has two methods for enabling the AKD. 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.

12.8.2 Disable Modes

Use DRV.DISMODE to select the method for stopping the drive.



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.TBRAKEAPP. 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.

Note: for Modes 2 and 3, you can access the Controlled stop section to set the values of the controlled stop deceleration rate, the velocity threshold, and time in velocity threshold to deactivate the drive.

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) until one of two things happens. 1) The motor reaches the velocity limit set (CS.VTHRESH) for a period of time (CS.TO), or 2) The drive emergency time out is reached (DRV.DSTO). 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.

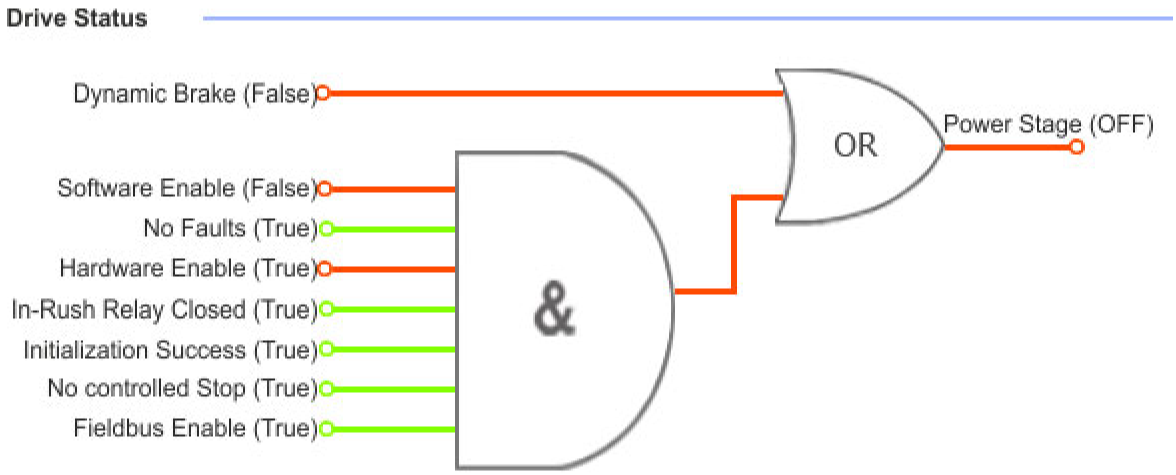
Similar to Mode 2, the motor will decelerate at a controlled rate (CS.DEC) until CS.VTHRESH is reached for a period of time (CS.TO). The drive will then dynamically brake and disable under the same conditions as described in Mode 2.

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.

12.8.3 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 in finding which input may be preventing the drive from being enabled. Click **More** to see the details of how the control stop is executed displayed both logically and graphically.



12.8.4 Controlled Stop

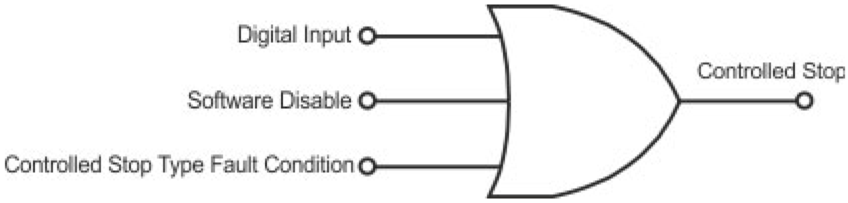
The Controlled Stop area displays values for parameters associated with controlled stop.

Button or Dialog Box	Description	Parameter
Velocity Threshold	Sets the velocity threshold for controlled stop process.	CS.VTHRESH (pg 452)
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 (pg 451)
Deceleration	Sets the deceleration value for the controlled stop process.	CS.DEC (pg 448)
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**. **Configure Input** link will opens the Digital I/O screen where you can configure the controlled stop mode.	

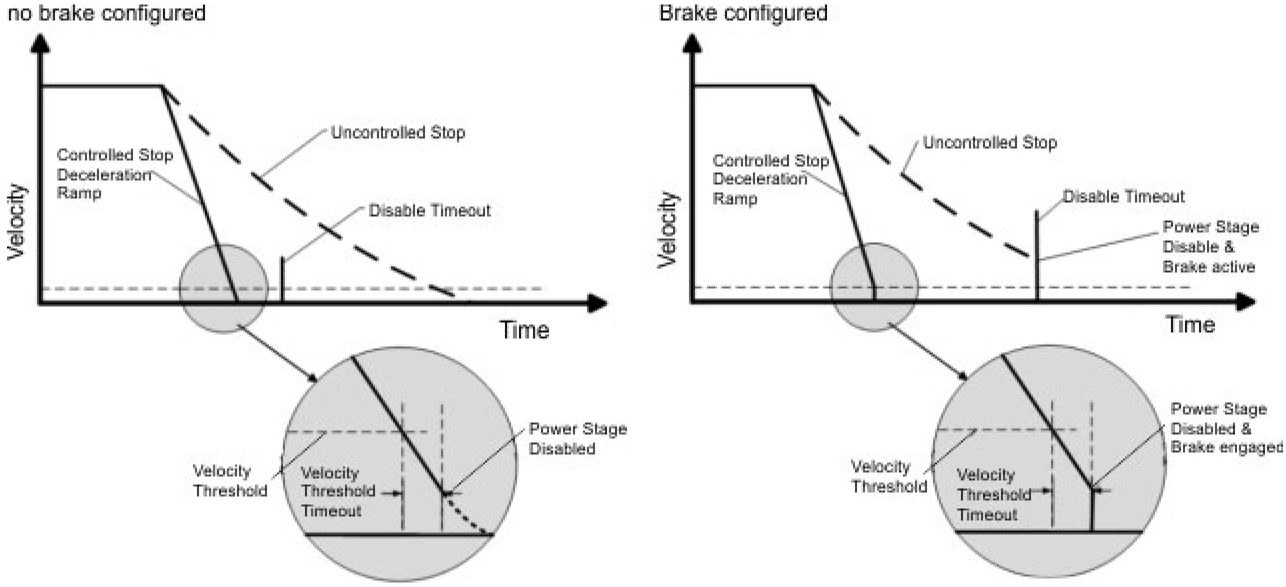
12.8.5 More/Less Button

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 available: one for a brake fitted and another for no brake.

Control stop status diagram



Controlled Stop Block Diagram



12.9 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 (pg 448)).

A controlled stop can occur in four 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 WorkBench terminal window) initiates a software disable (DRV.DIS) command .
- CANopen PDO is set to 3442.

CANopen Property	Value
Index/Subindex	3442/0
Data Type	Unsigned 8
Access	W/O
PDO mappable	N/A
Description	Controlled stop
ASCII Object	

- A fault initiates a controlled stop from the drive. See Fault and Warning Messages (pg 250) for the faults which initiate a controlled stop.

The controlled stop mechanism is activated in the following cases:

1. DRV.DISMODE = 2 and user executes DRV.DIS from the terminal or WorkBench disable buttons.

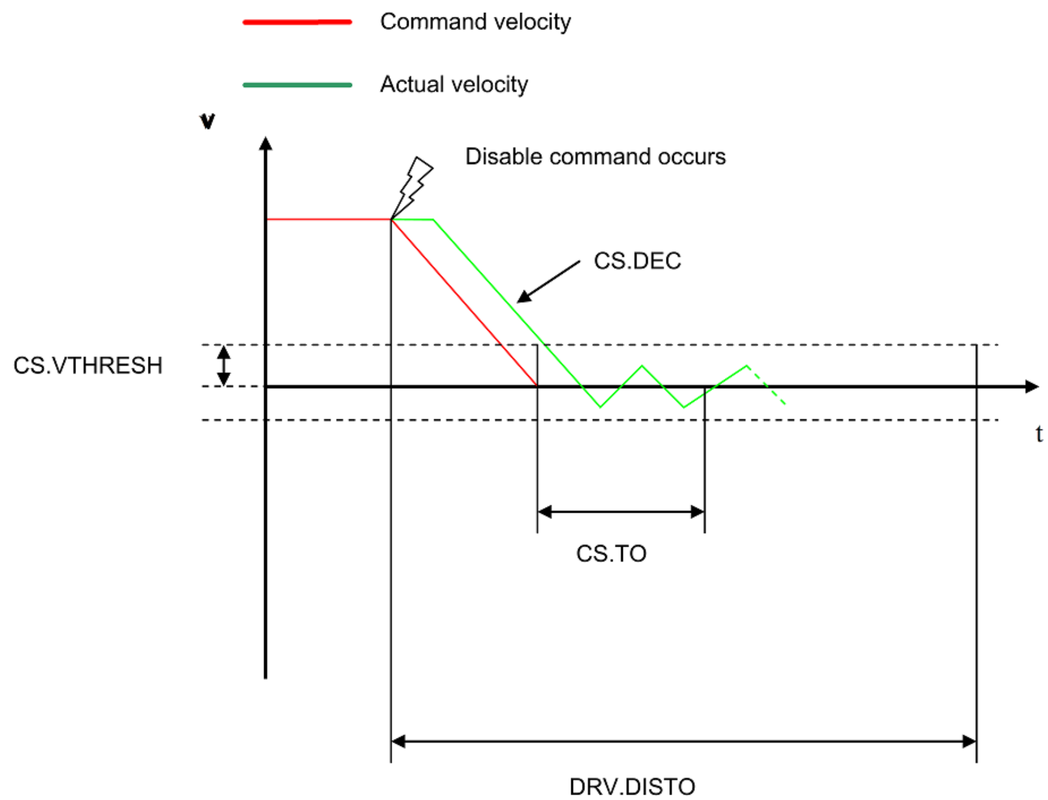
NOTE You must disable the drive in order to set DRV.DISMODE.

2. DRV.DISMODE = 2 and user executes DRV.DIS from a fieldbus connected to the drive.
3. A fault happens for which the reaction evolves controlled stop (CS). After the CS is executed, the drive disables.
4. 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.
5. 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.DISTO is not active.
6. 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.DISTO 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 ± CS.VTHRESH.
3. CS.TO: Velocity 0 time. The actual velocity must be consecutively within 0 ± 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 VEL0 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 VEL0 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 ± CS.VTHRESH.

The drive issues a fault FF703 (pg 263) in case that the DRV.DISTO counter expires during a controlled stop procedure.

Related Parameters and Commands

CS Parameters (pg 447)

CS.STATE (pg 450): Reads the current state of controlled stop process (0 = controlled stop is not occurring. 1 = controlled stop is occurring).

DIN1.MODE TO DIN24.MODE (pg 461)

DRV.DIS (pg 506)

DRV.DISTO (pg 511)

DRV.DISMODE (pg 507)

Related topics:

Emergency Stop (pg 128)

Digital Inputs and Outputs (pg 100)

Fault and Warning Messages (pg 250)

12.10 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.

12.10.1 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.

AKD-x00306 to AKD-x00606

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.

AKD-x01206 to AKD-x02406 and AKD-xzzz07

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 Installation Manual* for detailed information on dynamic braking.

DRV.DISMODE (pg 507)

DRV.DBILIMIT (pg 500)

12.11 Emergency Stop

12.11.1 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.

NOTE The parameter DRV.DISMODE must be set to 2 to implement the different stop categories. Consult the *AKD User Guide* for configuring the parameter.

WARNING

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.

Set parameter MOTOR.BRAKEIMM to 1 with vertical axes, to apply the motor holding brake immediately after faults or Hardware Disable.

12.11.1.1 Stop

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

NOTE 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 61508 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.

12.11.1.2 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).

NOTE 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.

12.11.1.3 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.

NOTE 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).

12.12 Safe Torque Off (STO)

Safe torque off (STO) is a restart lock safety feature that prevents a system restart. The STO input provides direct electrical control of the power stage of the AKD drive; it bypasses the processor and will disable the power stage irrespective of software or other hardware signals.

STO is controlled by a digital input on the X1 connector (pin 3) that must have 24V applied to it; otherwise, the drive will not enable.If the STO digital input does not have 24V applied to it and you attempt to enable the drive (using the hardware and software enables) the drive will generate a 602 “Safe Torque Off” fault. If you see this fault you need to apply 24V to the STO input and then clear the fault (DRV.CLRFAULTS) before you can enable the drive. The STO will not generate a fault until you try to enable the drive. You can read the current faults using DRV.FAULTS.

```
-->DRV.FAULTS
602: Safe torque off.
-->
```

The current state of the STO can be read using the STO.STATE parameter (this returns 1 if 24V is being applied to this input). WorkBench also shows you the state of the STO input in the status bar at the bottom of the window.

12.13 Under Voltage Fault Behavior

You can adjust the conditions for an under voltage fault using VBUS.UVMODE in the WorkBench Terminal (pg 240) 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.

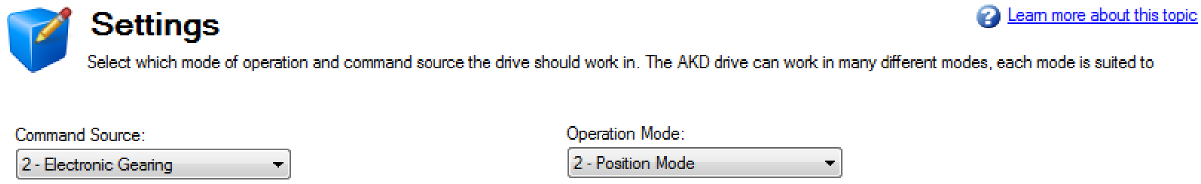
When an under voltage fault occurs, the drive is disabled and issues the following alerts:

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

13 Using Command Source and Operating Modes

13.1 Overview

Operation modes (opmodes) allow you to set up your drive to communicate directly over the Ethernet input, a specific fieldbus, or an analog or digital control.



There are two basic components to how you will command the drive and how it will behave. The “Service mode” indicates how the drive will be communicated to. The drive has communication options via Ethernet, a variety of fieldbuses, through an analog input, and through electronic gearing, or digital inputs. The second component ties to which loop you will be controlling (torque, velocity, or position).

13.2 Using Command Source and Operation Modes

There are two methods to access these two parameters within WorkBench. The first is by selecting the **Settings** screen from the left hand tree. By clicking on the top level of the **Settings** folder, you access the graphical representation of both the Command source and Operation Mode. The drop-down box then allows you to select the desired command type and control loop you wish to activate. Please note that some Command Sources can only be used with certain control loops (as an example, Electronic Gearing can only be used in position loop Operation Mode).

13.2.1 Command Source

The command source sets how you communicate with the drive. Initially, you might be communicating via your PC using the Ethernet connection. Each Command Source is listed below:

13.2.1.1 Service

This is the most common source used when establishing initial communications with the drive to set the system up, and when you may need to “service” the drive. The Service source communicates with your PC via the Ethernet port located on the top of the drive at connector X11.

13.2.1.2 Fieldbus

When using a fieldbus, such as CANOpen or EtherCAT, the drive is set with this command source. For CANOpen, use the connectors X12 and X13 located on the top of the drive. For EtherCAT, Connectors X5 and X6 located on the front of the drive are used.

13.2.1.3 Electronic Gearing

If the drive will be used to follow the output of an external encoder, following motion using an electronic gear ratio, then you should use this command source. When using Electronic Gearing, the Operation mode must be set in the Position Loop mode. This is also the mode used for step and direction inputs as well.

13.2.1.4 Analog

This command source allows the drive to be controlled from an analog source. Typically a +/- 10 Vdc signal is attached to connector X8 pins 9 and 10. Varying the analog input will then vary torque, velocity, or position based on the Operation mode selected.

13.2.2 Operation Mode

The operation mode identifies which servo loop you will be controlling. The drive offers torque, velocity, or position control. From the setting screen page, a graphical representation of the loop is displayed. By clicking these graphical representations, you can access additional loop information such as gains, filters, and other settings.

Related Parameters

DRV.CMDSOURCE (pg 497)

DRV.OPMODE (pg 549)

13.3 Current Loop

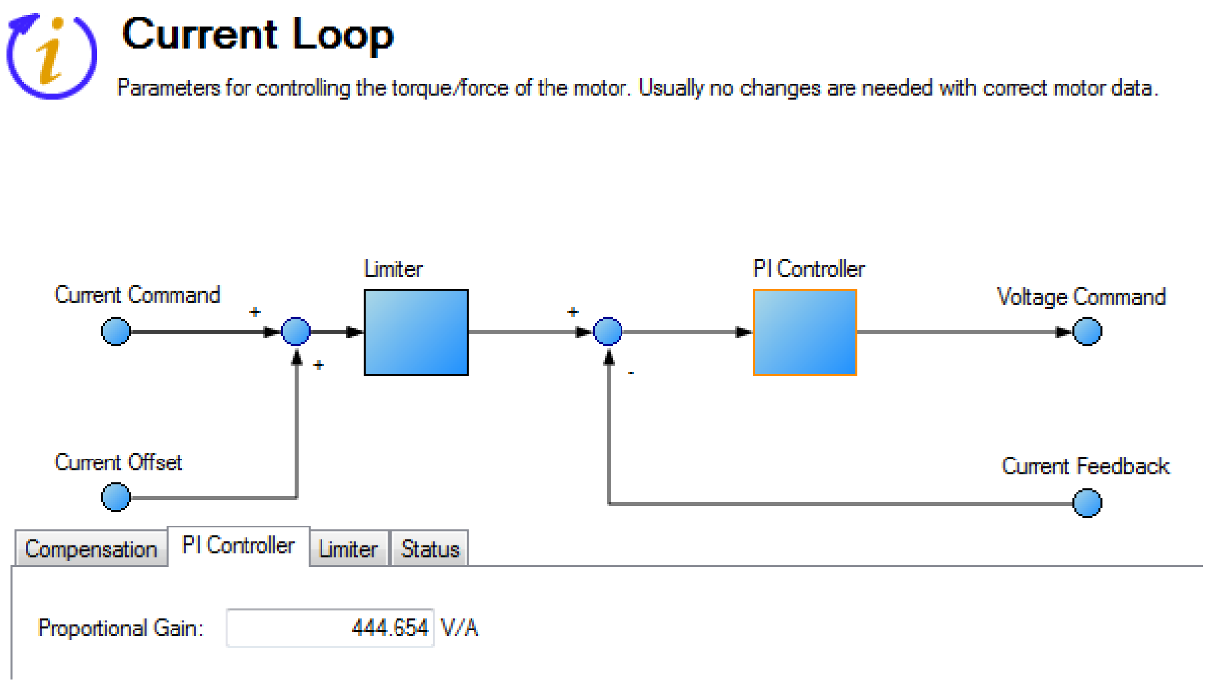
13.3.1 Overview

The current loop is active when the drive operates in current torque (current) mode (DRV.OPMODE (pg 549) = 0). The parameters that govern the current loop are shown in the Current Loop view. The various types of tuning for the drive adjust these parameters automatically, so you normally do not need to adjust the current loop parameters in the current loop screen. The Current Loop view includes an active block diagram. If you click on a block in the diagram, the appropriate tab opens below.

A more detailed block diagram for the current loop is included in Block Diagrams (pg 308)

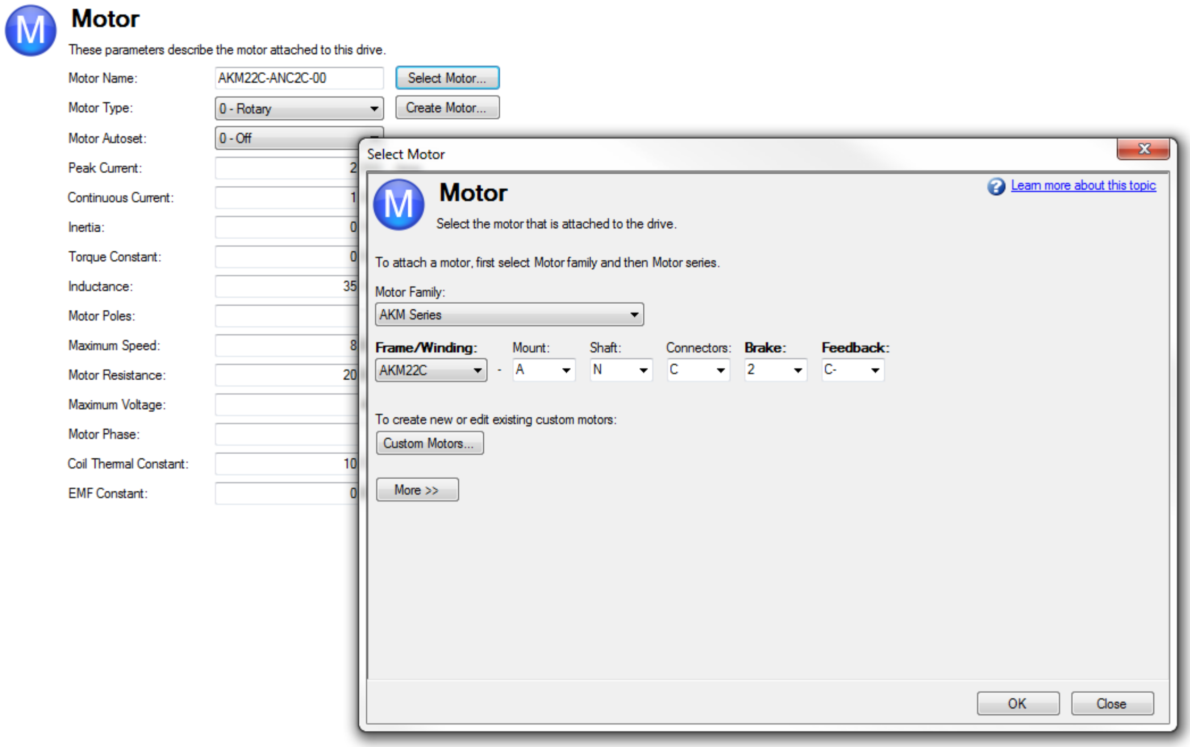
13.3.2 Current Loop Gain

The current loop is tuned based on the inductance of the motor used with the drive. The current loop gain is automatically set so that the idealized current loop crossover frequency is IL.KP/L in rad/sec where L is the motor line-line inductance.



Current loop gain is set automatically by the drive using the following methods:

- **Option A**. When a feedback device is automatically identified by the drive and the motor data is automatically populated (Motor Autoset = 1-On) the proportional gain of the current loop (il.kp) is set based on the motor data and is shown as a read-only parameter in the current loop screen.
- **Option B**.When the motor is selected using the motor database or using the custom motor tool, the imported inductance value is used to set the current loop proportional gain.



NOTE Manual adjustments to the current loop proportional gain parameter are not normally required during the motor tuning procedure. If manual adjustments are made to the current loop proportional gain parameter, repeating the motor setup procedure will overwrite the changes and restore the value to the Kollmorgen calculated value.

Related Parameters

IL Parameters (pg 684)

DRV.OPMODE (pg 549)

13.3.3 Current Loop Gain Scheduling

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 (pg 755))

To rectify this problem, you can use gain scheduling to change the current loop gain as a function of the current command (IL.CMD).

13.3.3.1 Using the Gain Scheduling View in WorkBench

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.

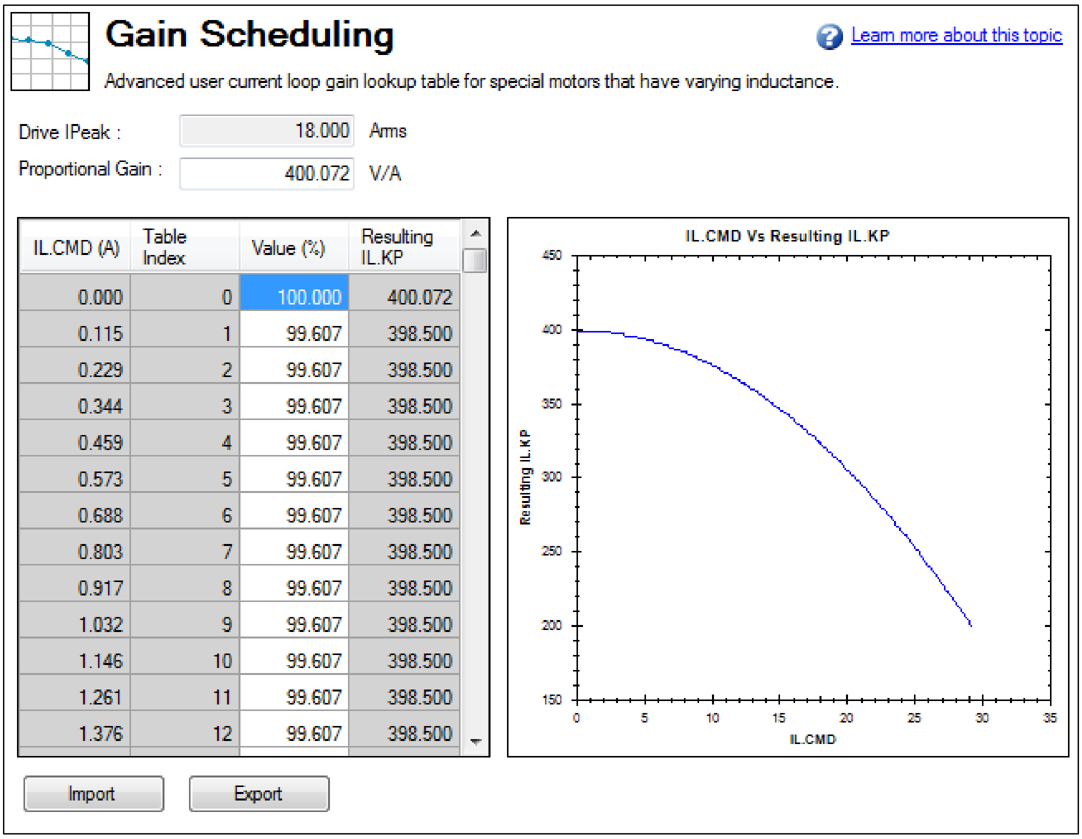
Current Loop Gain = 2000 * 2π* Motor Inductance (H)

Example

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

Current Loop Gain = 2000 * 2 π * 0.00319 = 40.01

Once suitable values have been identified, you can enter these values into WorkBench 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.



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

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 (pg 705) and IL.KPLOOKUPVALUE (pg 706). 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 Range = DRV.IPEAK/157 * IL.KPLOOKUPINDEX

A full list of table values can also be retrieved using IL.KPLOOKUPVALUES (pg 707), 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
```

13.4 Velocity Loop

13.4.1 Overview

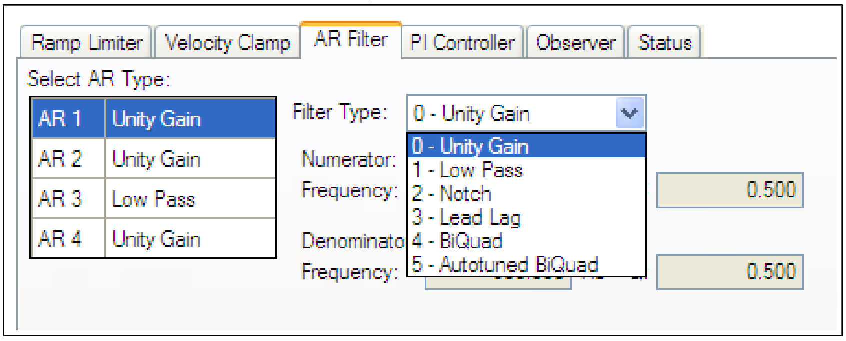
The velocity loop is active when the drive operates in velocity mode (DRV.OPMODE (pg 549) = 1) or position mode (DRV.OPMODE = 2). The parameters that govern the velocity loop are shown in the Velocity Loop view. This view is only available while in Operation Mode 1 or 2 (set in the Settings view). The various types of tuning for the drive adjust these parameters automatically, so you normally do not need to adjust the velocity loop parameters in the velocity loop screen.

A detailed block diagram for the velocity loop is included in Block Diagrams (pg 308).

13.4.2 Tabs in the Velocity Loop View

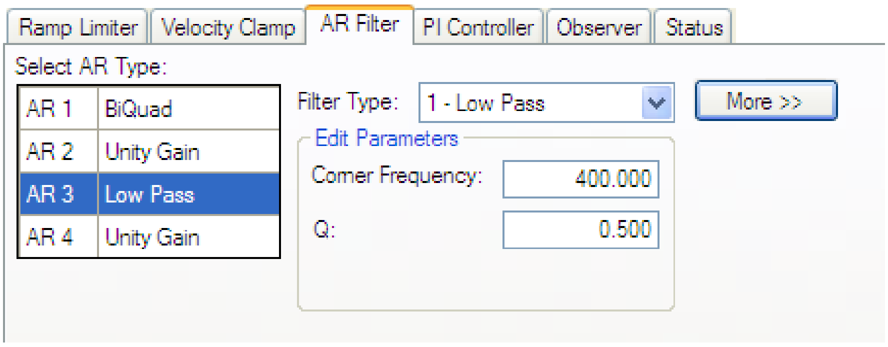
The velocity view includes an active block diagram. If you click on a block in the diagram, the appropriate tab opens below.

- **Ramp limiter** . The ramp limiter consists of the acceleration limits of the drive. These acceleration limits override both motion task and electronic gearing acceleration limits, so they must be set higher than the highest required motion task acceleration or gearing acceleration value. These acceleration and deceleration limits are also shown in the Service Motion view and the Limits view (DRV.ACC (pg 489) and DRV.DEC (pg 501)).
- **Velocity clamp**. The velocity clamp affects the maximum speed of the drive when the command source is service (DRV.CMDSOURCE (pg 497) = 0). This speed limit affects motion commanded in service motion and in motion tasks. These limits are also found in the limit screen on WorkBench. (VL.LIMITP (pg 949) and VL.LIMITN (pg 947))
- **AR1, AR2, AR3, AR4**: These values are the independent bilinear quadratic (bi-quad) filters inside the drive. AR1 and AR2 are in the forward path and AR3 and AR4 are in the feedback path. These bi-quad filters can each be configured in five different modes.



0–Unity Gain. The filter is off, and it will not affect the loop.

1–Low Pass. In modes 1, 2, and 3, the bi-quad filter is configured for each respective type of filtering. The Edit Parameters field is used to set up the filter. The actual bi-quad filter values are shown to the left:



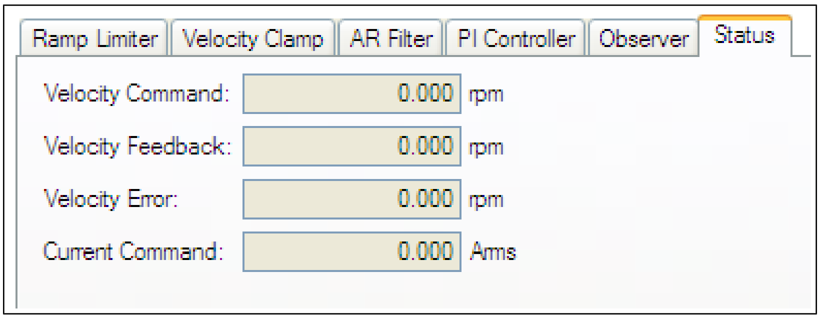
2–Notch

3–Lead Lag

4–Bi-quad . A manually configured Bi-quad filter. This is an advanced tuning function.

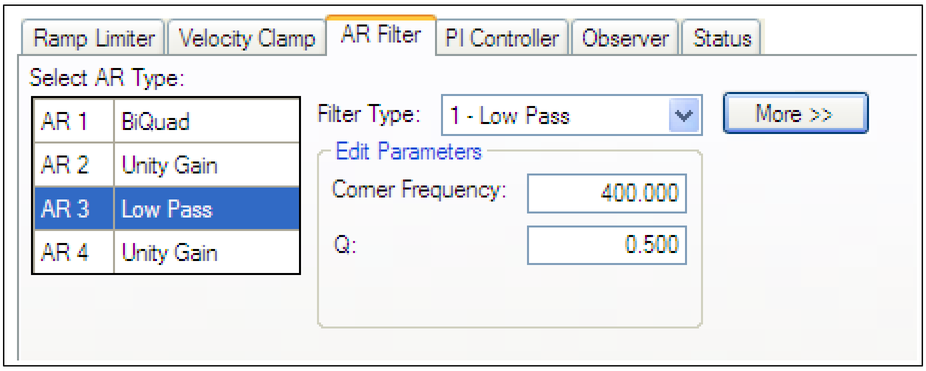
5–Autotuned Biquad . When the PST sets a filter after the PST process is complete, the values are input into the Bi-Quad filter and are shown as read only values.

- **Status**. The status tab shows parameters that are relevant to the velocity loop performance.



13.4.3 Velocity Loop Default Settings and Changes

By default, a PI loop with a low-pass filter (AR3) is set in the drive.



The default value for the low-pass filter is 400 Hz. The low-pass filter is important for disturbance rejection, and it also reduces the audible noise of the system.

Velocity Loop Changes Based on Slider Tuning

Slider tuning (see Slider Tuning (pg 183)) uses the slider control to adjust the proportional gain and integral gain values of the velocity loop based on the desired bandwidth. If you adjust the bandwidth using the slider tuner and then return to the velocity loop screen, you will see different values inside the proportional gain and integral gain fields. No adjustment is made to the filters automatically by using the slider tuner. Only the proportional and integral terms are adjusted.

Velocity Loop Changes Based on PST

When the PST (see Using the Performance Servo Tuner (pg 183)) is used, changes are made to the proportional gain, integral gain, filters, and other parameters not related to the velocity loop screen directly. The values adjusted are dependent on the drive, motor, load, and the PST settings. The filters that are adjusted by the PST are automatically put into mode **5-Autotuned Bi-quad**.

No adjustments can be made to mode **5- Autotuned BiQuad** filters that are set by the PST. If adjustment to the tuning of the system is desired after the PST process is completed, then these adjustments should be made in the settings of the PST. The PST process can then be repeated.

13.4.4 Biquad Filters

Filters in the AKD all exist as digital biquad filters in the servo loops. Lowpass, LeadLag, and Resonator filters are derived by the following equations. WorkBench handles all the math involved for the user. Enter the values in the fields for the type of filter desired.

Generate a Biquad as a Lowpass at Frequency F

Numerator Frequency = 5000

Numerator Q = Sqrt(2)/2 (this is 0.707)

Denominator Frequency = **F**

Denominator Q = Sqrt(2)/2 (this is 0.707)

Generate a Biquad as a LeadLag at Frequency F, Gain G

Numerator Frequency = **F** * 10^(-**G**/80)

Numerator Q = Sqrt(2)/2 (this is 0.707)

Denominator Frequency = **F** * 10^(**G**/80)

Denominator Q = Sqrt(2)/2 (this is 0.707)

Generate a Biquad as a Resonator at Frequency F, Gain G, Bandwidth Q

Numerator Frequency = **F**

Numerator Q = 10^(-**G**/40) * **Q**

Denominator Frequency = **F**

Denominator Q = 10^(**G**/40) * **Q**

Related Parameters

VL Parameters (pg 920) | DRV.ACC (pg 489) | DRV.CMDSOURCE (pg 497) | DRV.DEC (pg 501) | DRV.OPMODE (pg 549)

Related Topics

Limits (pg 119) | Service Motion (pg 164) | Tuning Your System (pg 182)

13.5 Position Loop

13.5.1 Overview

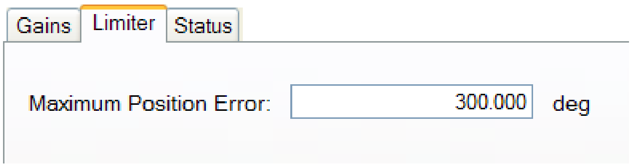
The position loop is active when the drive operates in position mode (DRV.OPMODE (pg 549) = 2). The parameters that govern the position loop are shown in the Position Loop view. The various types of tuning inside AKD adjust these parameters, so you normally do not need to adjust the position loop parameters in the position loop screen.

A detailed block diagram for the position loop is included in Block Diagrams (pg 308).

13.5.2 Tabs in the Position Loop View

The position loop view includes an active block diagram. If you click on a block in the diagram, the appropriate tab opens below.

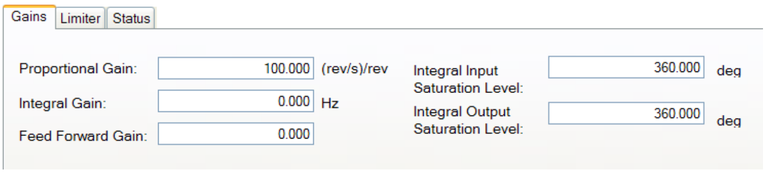
- **Gains**. This tab shows the gains for the position loop.
- **Limiter**. The value in the **Maximum Position Error** box (PL.ERRFTHRESH (pg 811)) limits the position error (PL.ERR (pg 810)) that can be present. When the maximum position error is exceeded, the drive generates fault F439 (pg 258), Following Error. If the maximum position error is set to 0 (default) then the maximum position error is ignored.



- **Status**. This tab shows the present value of commanded position (PL.CMD (pg 809)), position feedback (PL.FB (pg 817)), position error (PL.ERR (pg 810)), and velocity command (VL.CMD (pg 931)).

13.5.3 Position Loop Default Behavior and Changes

By default, only a proportional gain (PL.KP (pg 825)) is applied in the position loop.



Position Loop Changes Based on Slider Tuning

Slider Tuning (see Slider Tuning (pg 183)) adjusts the proportional gain of the position loop (along with velocity loop view parameters; see Velocity Loop (pg 137)). If you adjust the bandwidth using the slider tuner, then when you return to the position loop screen, you will see a change to the proportional gain only. No adjustment is made to the integral gain or feedforward gain through the slider tuner. The integral saturation levels are not applicable when the integral gain is set to 0. In the **Gains** tab, the boxes for these values may be populated with default values whether or not the integral gain is set to 0.

Position Loop Changes Based on PST

When the Performance Servo Tuner (PST, see Using the Performance Servo Tuner (pg 183)) is used, changes are made to the position loop proportional gain, integral gain, feedforward gain and other parameters not related to the Position Loop view directly. The values adjusted are dependent on the drive, motor, load, and the PST settings.

Related Parameters

PL Parameters (pg 808)

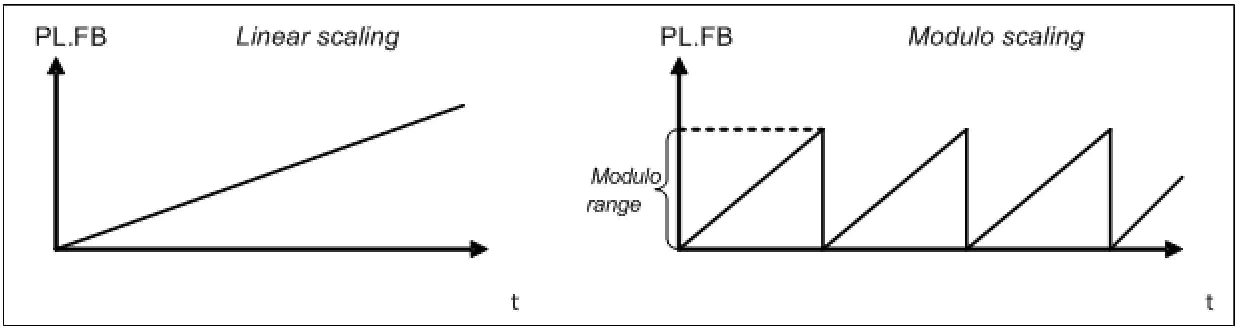
DRV.OPMODE (pg 549)

VL.CMD (pg 931)

13.5.4 Modulo Position

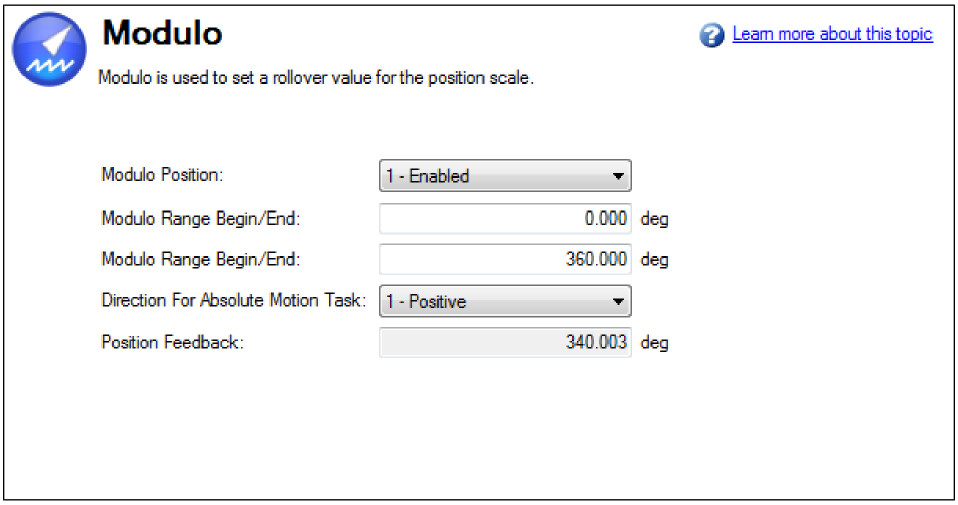
Modulo position is a function that simplifies rotational applications such as unidirectional rotating assembly tables. When enabled, the modulo axis feature converts several position-based parameters to fit in a defined modulo range. Once this range is defined, a given position value will roll over at the end of the modulo range and return to the beginning of the modulo range. This behavior affects some drive functions, which work with modulo-scaled position variables when the modulo-feature is enabled.

The following figure describes the progress of the actual position value (PL.FB) for linear scaling and modulo scaling when the motor moves continuously in a positive direction:



13.5.4.1 Setting up the modulo axis in WorkBench

You can set up the modulo axis from the **Modulo** view in WorkBench.



Button or Box	Description
Modulo Position	Enables or disables modulo (PL.MODPEN (pg 829)
Modulo Range Begin/End	Sets the beginning and end of the modulo range (PL.MODP1 (pg 826), PL.MODP2 (pg 827))
Direction for Absolute Motion Task	Sets the direction of an Absolute Motion task when Modulo is enabled. The direction can be set to always positive, or always move negative. The "Shortest Distance" mode will determine the shortest distance to the target and move in that direction. The "inside range" mode will move in the direction that allows the motor to stay in between the defined scale and therefore not to wrap around. Shortest Distance is more commonly used than Inside Range mode. (PL.MODPDIR (pg 828))
Position Feedback	Reads and displays the position feedback (PL.FB (pg 817))

13.5.4.2 Setting up the modulo axis from the Terminal

You can use the following parameters to configure the modulo-axis feature:

- PL.MODPEN (pg 829): Enables or disables the modulo-axis feature.
- PL.MODP1 (pg 826): Defines either the beginning or the end of the modulo range, depending on the PL.MODP2 setting.
- PL.MODP2 (pg 827): Defines either the beginning or the end of the modulo range, depending on the PL.MODP1 setting.

13.5.4.3 Parameters affected by the modulo axis

The following parameters are converted into modulo format when the values of these parameters are queried by a user, a fieldbus, or the software oscilloscope.

- PL.FB (pg 817): The actual position of the drive is converted into modulo scaling.
- PL.CMD (pg 809): The command position of the drive is converted into modulo scaling.
- CAP0.PLFB (CAP0.PLFB, CAP1.PLFB (pg 438)): The actual position of the drive, which has been captured by the capture engine 0, is converted into modulo scaling.
- CAP1.PLFB (CAP0.PLFB, CAP1.PLFB (pg 438)): The actual position of the drive, which has been captured by the capture-engine 1, is converted into modulo scaling.

13.5.4.4 Drive functions affected by modulo axis

Software limit switch

The software limit switches in the drive compare the actual position (PL.FB (pg 817)) with threshold values. Motion is stopped when the actual position exceeds the software limits. Since PL.FB is affected by the modulo-axis feature, the software limit switches monitor the modulo-converted PL.FB value. Software limit switches with thresholds outside of the modulo range never limit motion.

Programmable limit switch

The programmable limit switches compare the actual position (PL.FB (pg 817)) with selectable thresholds and then status flags are set to "true" when the actual position is currently within these position limits. The programmable limit switches monitor the modulo-converted PL.FB value. Programmable limit switches that are set outside the modulo-range never become active.

Digital output Modes 5 and 6

The digital output modes 5 and 6 (position greater than x, position less than x) compare the actual position of the drive with thresholds and activate the associated outputs when PL.FB (pg 817) is lower or higher than the threshold. The digital output mode functionality monitors the modulo-converted PL.FB value. Position thresholds that are set outside the modulo range continuously activate or deactivate the digital output.

Motion tasking to absolute target positions

When modulo is enabled, absolute motion tasks assume that the command is modulo converted. Absolute motion tasks to target positions outside of the modulo range generate a warning, nFault and Warning Messages (pg 250) (Motion task target position is out of modulo range).

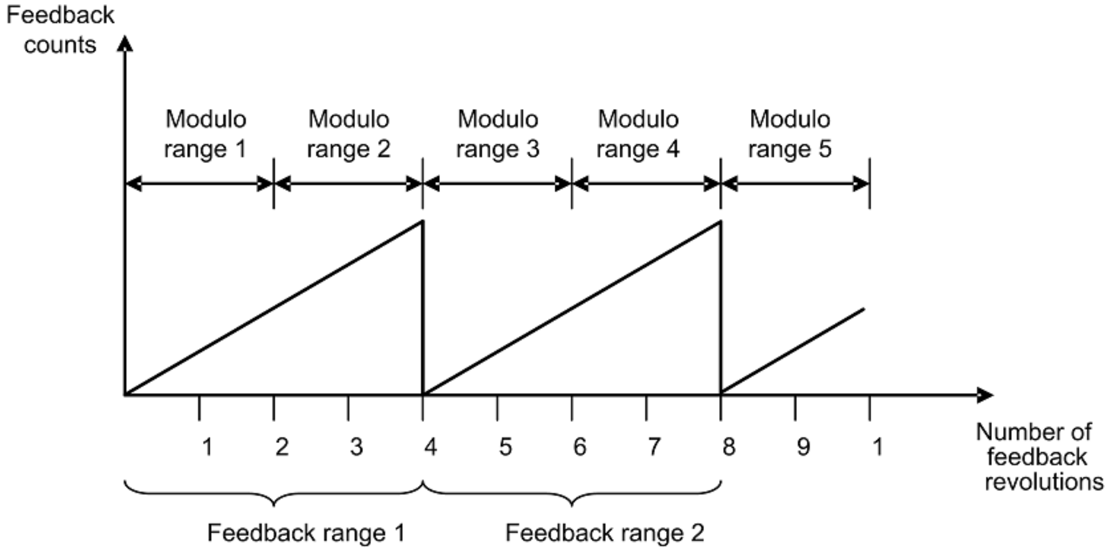
13.5.4.5 Using the modulo position feature with multiturn encoders

A special case exists for the following combinations of events:

- The drive is connected to a multiturn feedback device.
- The modulo axis feature is enabled.
- The selected modulo range does not fit as an integer in the range of the multiturn feedback.
- The application moves further than the total amount of multiturn feedback revolutions. In this case, problems occur because the multiturn feedback position overrun and a modulo range position rollover point does not occur at exactly the same position.

After powerup of the drive, the actual position (PL.FB (pg 817)) will be read from the multiturn feedback device. This position can be considered as a position within the feedback range as described in the figures below.

The following figure illustrates drive behavior when the selected modulo range fits as an integer within the multiturn feedback range. For simplicity, assume that one multiturn feedback range describes four feedback revolutions and the selected modulo range is set to two feedback revolutions.



As described in the figure above, the selected modulo range of the drive is repeated exactly at the point where the connected multiturn feedback rolls over (Modulo range 1, 3, 5…). The application can move for several multiturn feedback ranges and the drive can recalculate the modulo position correctly after a power cycle. The positions within the modulo ranges represent the same value in modulo format for each feedback range.

Example

The modulo-converted position, which represents 5 or 9 feedback revolutions, corresponds to the modulo position, which represents 1 feedback revolution.

The next figure illustrates the drive behavior when the selected modulo range does not fit as an integer within the multiturn feedback range. For simplicity, assume that one multiturn feedback range describes four feedback revolutions and the selected modulo range is set to 2.5 feedback revolutions.

As described in the figure above, the selected modulo range is not repeated exactly at the place where the connected multiturn feedback rolls over. The application can move for several multiturn feedback ranges, but the drive cannot calculate the modulo position correctly after a power cycle.

Example

The modulo-converted position, which represents five feedback revolutions, does not correspond to the modulo-position, which represents one feedback revolution.

14 Creating Motion

14.1 Homing

14.1.1 Overview

Homing is used to mechanically move a motor (connected to a mechanism) to a specific location on the machine, referred to as “home”. Motion tasks then use this home as a reference point for movements that must base a known position on this reference point. Motor movement is usually controlled by a variety of limit switches (end of travel) and a home reference switch. Using these reference points with the logic of the drive allows the machine to find and set the home reference point.

14.1.2 Using Homing

The AKD includes a variety of homing methods (set with HOME.MODE (pg 672)) to accommodate your machine needs:

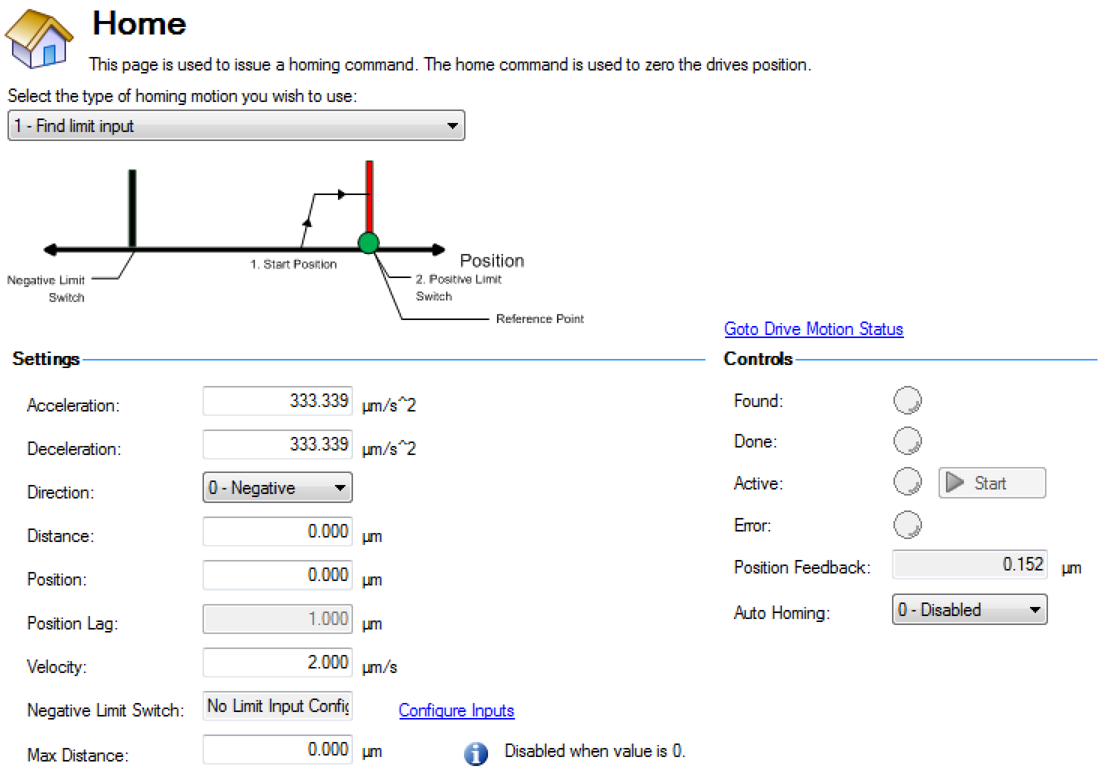
- Home using current position (HOME.MODE 0)
- Find limit input (HOME.MODE 1)
- Find input limit then find zero angle (HOME.MODE 2)
- Find input limit then find index (HOME.MODE 3)
- Find home input (HOME.MODE 4)
- Find home input then find zero angle (HOME.MODE 5)
- Find home input then find index (HOME.MODE 6)
- Find zero angle (HOME.MODE 7)
- Move until position error exceeded (HOME.MODE 8)
- Move until position error exceeded, then find zero angle (HOME.MODE 9)

Each of these homing methods offers a different way to achieve a home reference point based on your particular system mechanics. All homing methods provide the options of adjusting the acceleration, deceleration, and speed for homing moves. In addition, once the homing move is completed, you can either set an offset position or make an offset move as required. Homing modes, guidance for mode selection, and homing examples are included in Selecting and Using Homing Modes (pg 148).

NOTE When using any of the methods that use homing switches and limits, please refer to the Input/Output section for proper wiring techniques.

Home Default Window

The Home window provides a means to select your homing method and configure the homing settings. This window also provides a simple controls to start homing and confirm homing success.



Mode Selection:

Use this box to select the appropriate homing mode. Homing modes are described below in Selecting and Using Homing Modes. The active options in the **Settings** area change depending on the homing mode selected.

Settings:

- **Acceleration**: Sets the acceleration ramp used during the homing procedure.
- **Deceleration**: Sets the deceleration ramp used during the homing procedure.
- **Direction**: Sets the start direction for homing movement.
- **Distance**: Sets a prescribed distance you want the motor to move once the home reference point is found. A zero value (default) corresponds to the axis actively returning to the defined position found during the homing process.
- **Position**: Sets the current position to a prescribed value once the home reference point is found.
- **Position Lag**: Sets the position error threshold, which is used for indicating home reference when using the hardstop modes 8 and 9.
- **Velocity**: Sets the initial velocity used for homing moves.
- **Velocity Factor**: In modes where a limit is reached, and direction is reversed, the velocity factor allows you to reduce the velocity as a precentage of the homing velocity.
- **Positive/Negative Limit Switch/Home Reference/Peak Current**: These fields appear based on the mode selected. For homing to limits and home reference, this field will indicate how the digital inputs are configured as well as provide a link to the digital input page. For Homing to a hard stop, the Peak Current field allows you to set the peak current limit desired during homing.

Controls:

- **Found**: When the home reference is found, then this indicator is green.
- **Done**: When the home move is complete, this indicator is green.
- **Active:** This indicator will be green while the Home move is taking place.
- **Error**:This indicator will be red if something in the homing sequence failed.

- **Position Feedback**: This window reports the current value for PL.PFB.
- Auto Homing: Allows the system to auto-home on power up.
- **Start/Stop**: Click this button to start or stop the selected homing method.

14.1.3 Selecting and Using Homing Modes

Homing Mode 0: Home Using Current Position

Using the current position is the most basic homing method. This method simply uses the current position of the motor as the home point reference. Two values allow you to further define homing with this method:

- Distance: A value other than zero will cause a movement of the motor the distance entered in counts (or other units based on your units settings). You can use this to establish a home point at some prescribed distance from the initial starting position of the motor. This home will be at the offset distance entered from zero.
- Position: You can use this parameter to set the value of the home position other then zero. This allows you to offset your home reference away from zero. PL.FB will be set to the value you enter when the motor reaches the home reference point (based on the method selected).

The distance and position offsets are available and behave similarly on all of the homing types. The motor will either move an additional distance (distance value) after it finishes the homing method, or set the position to the amount entered in the position value.

Homing Mode 0 Example

Use the current position as home and have the motor end motion 180 degrees from home:

1. Select Mode 0 from the drop-down box.
2. Enter 180 into the **Distance** box.
3. Click **Start**.
4. The motor will move 180 degrees from the start position. The **Position Feedback** box (PL.FB) will show 180 (the motor is now sitting 180 degrees from home).

Homing Mode 1: Find Limit Input

The find limit input mode creates a move to a limit input. This method can be used if you have a positive or negative limit switch available that you want to establish as a home reference point.

NOTE Limit switches should be set to Active Low (when switch power is lost, no current is flowing, and the drive homes at the point of the loss in power of the limit switch).

The sequence of this homing mode is as follows:

1. The motor starts to move in the positive or negative direction, depending on the value you set in the **Home** screen (in the **Settings** section, **Direction** box).
2. The motor stops as soon as the hardware limit switch has been detected and then reverses direction.
3. The home position is set when the limit switch is no longer active. The actual and command position of the drive is immediately set to the home position value (HOME.P) and the motor ramps down to zero velocity. The axis is then moved to the position (HOME.P) + distance offset (HOME.DIST).

The values for distance and position can be used as described in homing mode 0.

⚠CAUTION When homing to a limit switch, the limit switch must remain in the triggered state while the motor decelerates to zero and begins to reverse. A very low acceleration rate combined with a high approach velocity may overshoot the switch and cause it to become

CAUTION active. This action will cause a homing error fault.

Homing Mode 1 Example

Use the positive end of travel limit as home reference, and then set this position to be -20 degrees.

1. Select Mode 1 from the drop-down box and enter 20 into the Position box.
2. Set the direction to positive. When Start is selected, the motor will move until it encounters the positive end of travel switch.
3. As soon as the switch is triggered, the motor will reverse direction until the switch is no longer active.
4. As soon as the switch is no longer active, the position will be set to -20 degrees and the motor will ramp to 0. Depending on the velocity you are homing with, and the settings of the acceleration/deceleration ramps, the position feedback will be close to the position you entered.

Homing Mode 2: Find Input Limit then Find Zero Angle

Similar to the Find Input Limit method, the find input limit then find zero angle[1] mode follows the same steps, but upon completion of the move, it continues to move to find the zero angle reference of the motor.

NOTE Limit switches should be set to Active Low (when switch power is lost, no current is flowing, and the drive homes at the point of the loss in power of the limit switch).

The specific steps are as follows:

1. The motor starts to move according to the direction (HOME.DIR) setting.
2. The motor stops as soon as the hardware limit switch has been detected and changes direction of movement.
3. The home position has been found as soon as the hardware limit switch is no longer active. The actual and the command position of the drive will immediately be set to the HOME.P value plus distance to the mechanical zero angle of the feedback device according to the current direction.
4. The motor moves to the home position (HOME.P) plus the distance move offset is applied (if present), which is located at the mechanical zero-angle of the feedback.

The values for distance and position can be used as described in home mode 0.

CAUTION When homing to a limit switch, the limit switch must remain in the triggered state while the motor decelerates to zero and begins to reverse. A very low acceleration rate combined with a high approach velocity may overshoot the switch and cause it to become active. This action will cause a homing error fault.

Homing Mode 2 Example

Use the positive end of travel limit as home reference and then move to the zero angle of the motor

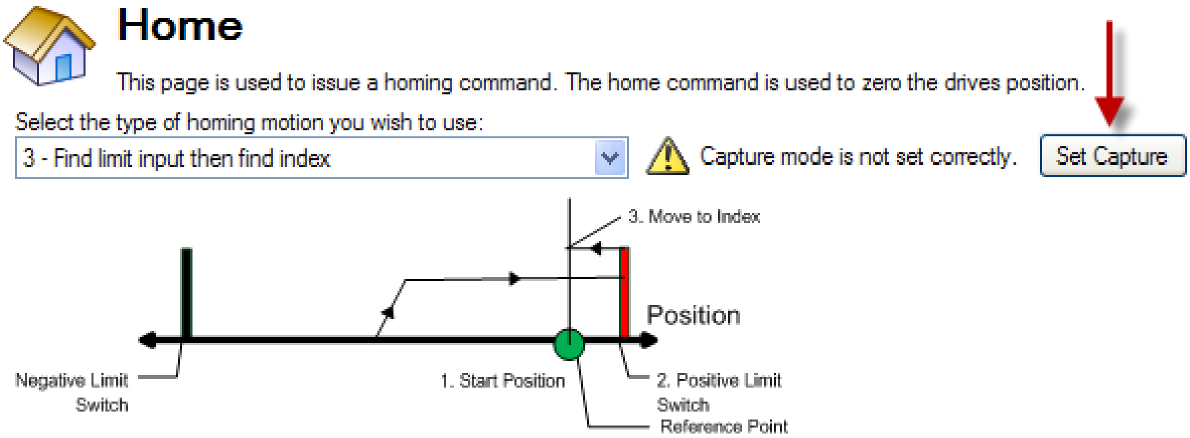
1. Use the positive end of travel limit as home reference and then move to the zero angle of the motor.
2. Select Mode 2 from the drop-down box.
3. Set the Direction to Positive.
4. When Start is selected, the motor will move until it encounters the positive end of travel switch.
5. As soon as the switch is triggered, the motor will reverse direction and move to the zero angle of the motor.

Homing Mode 3: Find Input Limit then Find Index

---

[1]Mechanical zero angle of the feedback = 0 degree.

Similar to the Find Input Limit method, this follows the same steps, but upon completion of the move, it continues to move to find the index pulse of the motor. This method can only be used with feedback devices that have an index pulse such as incremental encoders and analog sine encoders with an index channel (Feedback selection 10, 11, 20, 21). This method requires that Capture Mode is turned on in the home screen. With Mode 3 selected, a **Set Capture** button appears (see arrow below). Click **Set Capture** to set the Position Capture mechanism correctly for proper homing with an index pulse.



Once homing is triggered, the homing routine is performed as follows:

1. The motor starts to move according to the HOME.DIR setting.
2. The motor stops as soon as the hardware limit switch has been detected and changes direction of movement.
3. The motor ramps down to a reduced velocity as soon as the hardware limit switch is no longer active (please refer also to HOME.FEEDRATE). The drive is searching for the index-signal during this time. The home-position has been found as soon as the index-signal has been detected by the drive.
4. The actual and the command position of the drive will be set to the HOME.P value as soon as the index pulse is found. The drive then ramps down to velocity 0. The axis is then moved to the position (home.p) + distance offset (home.dist).

⚠CAUTION When homing to a limit switch, the limit switch must remain in the triggered state while the motor decelerates to zero and begins to reverse. A very low acceleration rate combined with a high approach velocity may overshoot the switch and cause it to become active. This action will cause a homing error fault.

Homing Mode 3 Example

Use the positive end of travel limit as home reference, and then move to the index reference of the motor feedback device at 50% of the original home velocity.

1. Select Mode 3 from the drop-down box.
2. Set the Direction to Positive.
3. In the **Home** screen click **Set Capture**.
4. Set the velocity factor to 50%.
5. When Start is selected, the motor will move until it encounters the positive end of travel switch. As soon as the switch is triggered, the motor will reverse direction, decelerate to a reduced velocity based on the Velocity Factor value, and move until the motor encounters the index pulses of the feedback device.

Homing Mode 4: Find Home Input

Homing Mode 4 establishes the home reference based on an external home switch connected to a drive digital input (DINx.MODE - 11 Home Reference).

The sequence of this homing mode is as follows:

1. The motor starts to move according to the direction (HOME.DIR) setting.
2. The home position has been found as soon as the home-switch becomes active while traveling in the selected direction of motion (HOME.DIR). The actual and command position of the drive will immediately be set to the position (HOME.P) value and the motor ramps down to velocity 0. The axis is then moved to the position (home.p) + distance offset (home.dist).

If the Home Reference Input is active when told to home, the drive does a reset and then the home sequence. The sequence of the reset:

1. The motor moves in the opposite direction of HOME.DIR
2. When the home switch is not active, the motor ramps down to zero, and subsequently follows the sequence of the homing mode.

The hardware limit switches are monitored during the homing procedure. The drive behaves as follows in case that a hardware limit switch is active before the home-switch has been activated:

a. The motor changes the direction until the home switch is crossed.
b. The motor ramps down to zero velocity and reverses direction again after crossing the home-switch.
c. The home-switch will now be activated according to the direction (HOME.DIR) setting and when the home-position has been found. The actual and the command position of the drive will immediately be set to the position (HOME.P) value and the motor ramps down to zero velocity. The axis is then moved to the position (home.p) + distance offset (home.dist).

Homing Mode 4 Example

Move in the negative direction towards the home reference point and then move 180 degrees from the reference point

1. Select Mode 4 from the drop down-box.
2. Set the **Direction** to **Negative** and enter 180 for distance.
3. Click **Start**.
4. The motor moves until it encounters the home reference switch. As soon as the switch is triggered, the motor moves an increment of 180 degrees as desired.

Homing Mode 5: Find Home Input then Find Zero Angle

The sequence of this homing mode is as follows:

1. The motor starts to move according to the Direction (HOME.DIR) setting.
2. The home position has been found as soon as the home-switch becomes active while traveling in the selected direction of motion (HOME.DIR) and the zero angle of the resolver has been found. The actual and the command position of the drive will immediately be set to the Position (HOME.P) value plus the distance to the mechanical zero angle of the feedback device according to the current direction.
3. The motor moves to the home Position (HOME.P) value plus the distance move offset is applied (if present), which is located at the mechanical zero-angle of the feedback.

If the Home Reference Input is active when told to home, the drive does a reset and then the home sequence. The sequence of the reset:

- The motor moves in the opposite direction of HOME.DIR
- When the home switch is not active, the motor ramps down to zero, and subsequently follows the sequence of the homing mode.

The hardware limit switches are monitored during the homing procedure. The drive behaves as follows in case that a hardware limit switch is active before the home-switch has been activated:

a. The motor changes the direction until the home switch is crossed.
b. The motor ramps down to zero velocity and changes afterwards the direction again after crossing the home-switch.
c. The home-switch will now be activated according to the HOME.DIR setting and when the home-position has been found. The actual and the command position of the drive will immediately be set to the position (HOME.P) value plus distance to the mechanical zero angle of the feedback device according to the current direction.
d. The motor moves to the home Position (HOME.P) value plus the distance move offset is applied (if present), which is located at the mechanical zero-angle of the motor feedback.

Homing Mode 5 Example

Move in the positive direction towards the home reference point and then move 60 degrees from the zero angle location.

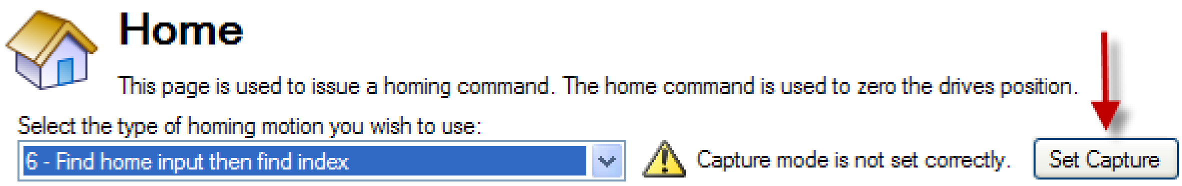
1. Select Mode 5 from the drop-down box.
2. Set the Direction to Positive and enter 60 for distance.
3. When Start is selected, the motor will move until it encounters the Home reference switch. As soon as the switch is triggered, the motor will move to the zero angle location plus an additional 60 degrees as desired.

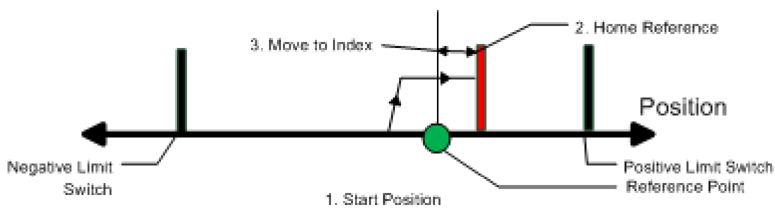
Homing Mode 6: Find Home Input then Find Index

Similar to the Home input method, this follows the same logic as the other homing methods, first completing the home to input method, and then finding the index pulse of the motor feedback.

This homing mode starts motion until a digital input, which is assigned to act as a home-switch, has been activated. The motor moves afterwards with a reduced velocity (HOME.FEEDRATE) until the index signal has been detected by the drive.

NOTE This method requires that Capture Mode is turned on. This is done in the home screen. With Mode 6 selected, a “Set Capture” button will appear (see arrow below). Pressing the button sets the Position Capture mechanism correctly for proper homing with an index pulse..





The home-switch must be activated according to the setting of the HOME.DIR setting.

The sequence of this homing mode is as follows:

1. The motor starts to move according to the HOME.DIR command.

2. The motor decelerates to a reduced velocity according to the HOME.FEEDRATE setting as soon as the home-switch becomes active during a motion in direction of the HOME.DIR setting.
3. The actual and the command position of the drive will immediately be set to the HOME.P value as soon as the index-signal has been detected. The motor decelerates until velocity 0 has been reached.

If the Home Reference Input is active when told to home, the drive does a reset and then the home sequence. The sequence of the reset:

1. The motor moves in the opposite direction of HOME.DIR
2. When the home switch is not active, the motor ramps down to zero, and subsequently follows the sequence of the homing mode.

The hardware limit switches are monitored during the whole homing procedure. The drive behaves as follows in case that a hardware limit switch is active before the home-switch has been activated:

a. The motor changes the direction until the home-switch is crossed.
b. The motor ramps down to zero velocity and changes direction again after crossing the home-switch.
c. The home-switch will now be activated according to the HOME.DIR command. The motor decelerates to a reduced velocity according to the HOME.FEEDRATE setting as soon as the home-switch becomes active.
d. The actual and the command position of the drive will immediately be set to the HOME.P value as soon as the index-signal has been detected. The motor decelerates until zero velocity has been reached. The axis is then moved to the position (HOME.P) + distance offset (HOME.DIST).

Homing Mode 7: Find Zero Angle

The sequence of this homing mode is as follows:

1. The home value is immediately found by the drive and the actual and the command position of the drive will immediately be set to the Position (HOME.P) value plus the distance to the mechanical zero angle of the feedback device according to the current direction.
2. The motor moves to the home Position (HOME.P) value, which is located at the mechanical zero-angle of the feedback.

Homing Mode 7 Example

Move in the positive direction towards the zero angle location.

1. Select Mode 7 from the drop-down box.
2. Set the direction to Positive.
3. When Start is selected, the motor will move to the zero angle location.

Homing mode 8: Move Until Position Error Exceeded

This method is also referred to as move to hard or mechanical stop. The AKD has several options related to this method as well. For this basic method, the motor will move until it encounters a hard stop, causing the position error to exceed a specific threshold you set. Once the threshold is exceeded, the motion stops and home reference is established. You can use the Distance or Position as described initially in this section.

⚠CAUTION Be sure to choose the direction appropriately to move off of the stop if using the distance offset.

The sequence in this homing mode is as follows:

1. As this home move is initiated, the motor will move according to the Direction (HOME.DIR) value until the positon error exceeds the Position Lag (HOME.PERRTHRESH) value.
2. The motor is now at home position (HOME.P) value.

Homing Mode 8 Example

Move in the positive direction towards a hard stop and limit the current to 1 amp. Allow 30 degrees of error before considered at the hard stop.

1. Select Mode 8 from the drop-down box.
2. Set the Direction to Positive, Position Lag to 30 degrees and Peak Current to 1.
3. When Start is selected, the motor will move to the hard stop with a peak current of 1 amp.
4. As the position error exceeds the 30 degrees, the home Position (HOME.P) is set.

Homing Mode 9: Move Until Position Error Exceeded then Find Zero Angle

The sequence in this homing mode is as follows:

1. As this home move is initiated, the motor will move according to the Direction (HOME.DIR) value until the positon error exceeds the Position Lag (HOME.PERRTHRESH) value.
2. The home value is immeadiately found by the drive and the actual and the command position of the drive will be set to the Position (HOME.P) value plus the distance to the mechanical zero angle of the feedback device according to the current direction.
3. The motor moves to the home Position (HOME.P) plus the distance move offset is applied (if present) value, which is located at the mechanical zero-angle of the feedback.

Homing Mode 9 Example

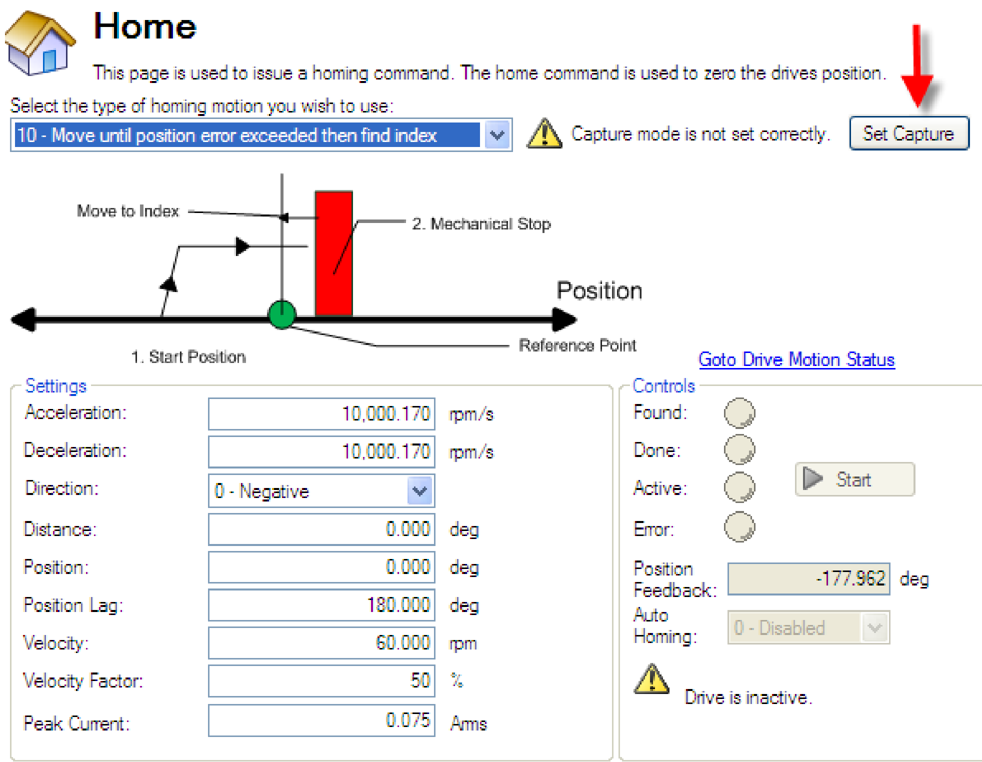
Move in the positive direction towards a hard stop and limit the current to 1 amp. Allow 30 degrees of error before considered at the hard stop. Then move to the zero angle of the motor and designate this as the 180 point.

1. Select Mode 9 from the drop-down box.
2. Set the Direction to Positive, Position to 180, Position Lag to 30 degrees and Peak Current to 1.
3. When Start is selected, the motor will move to the hard stop with a peak current of 1 amp. As the position error exceeds the 30 degrees, the home Position (HOME.P) is set and the Position will be set to 180.

Homing Mode 10: Move Until Position Error Exceeded then Find Index

This method is similar to HOME.MODE 8, but looks for the index pulse after it encounters the hard stop. For this method, the motor will move until it encounters a hard stop, causing the position error to exceed a specific threshold that you set. Once the threshold is exceeded, the motion will reverse and look for an index pulse.

This method can only be used with feedback devices that have an index pulse such as incremental encoders and analog sine encoders with an Index channel (Feedback selection 10, 11, 20, 21). This method requires that Capture Mode is turned on in the home screen. With Mode 10 selected, a **Set Capture** button appears (see arrow below). Click **Set Capture** to set the position capture mechanism correctly for proper homing with an index pulse.



1. Motor advances into the mechanical stop and then reverses direction.
2. The motor is searching for the index pulse during this time.
3. If the motor finds the Index pulse, then the Home-position is found.
4. The actual and the command position of the drive will be set to the HOME.P value as soon as the index signal is found. The drive then ramps down to velocity 0.
5. If another mechanical stop is found before the Index signal, then the homing sequence will fail and the system needs to be reviewed for proper wiring.

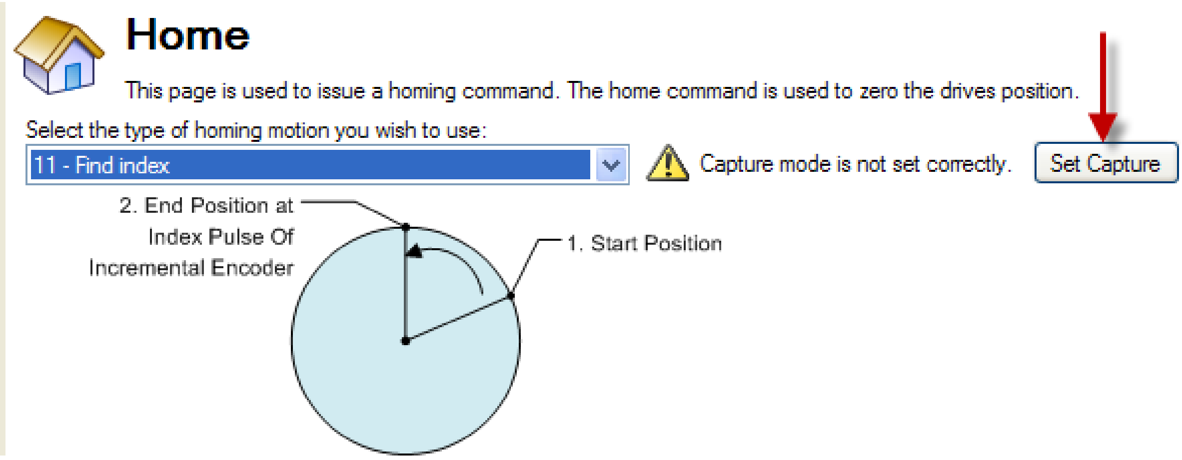
Homing Mode 10 Example: Move Until Position Error Exceeded then Find Index.

1. Select mode 10 from the drop-down box.
2. Set the direction to positive.
3. In the home screen click **Set Capture**.
4. Set position lag and peak current values based on your application requirements.
5. When start is selected, the motor moves in the positive direction until a hard stop is encountered.
6. The motor reverses and moves until it encounters the index reference and then it stops.
7. If another hard stop is encountered before the index reference, home fails.

Unlike Homing Mode 3, the Home position is set as soon as the index pulse is found, irrespective of the direction of motion.

Homing Mode 11: Find Index Signal

This method can only be used with feedback devices that have an index pulse such as incremental encoders and analog sine encoders with an Index channel (Feedback selection 10, 11, 20, 21). This method requires that Capture Mode is turned on in the home screen. With Mode 11 selected, a **Set Capture** button appears (see arrow below). Click **Set Capture** to set the position capture mechanism correctly for proper homing with an index pulse.



Once homing is triggered, the homing routine is performed as follows:

1. The motor starts to move according to the HOME.DIR setting.
2. The Motor is searching for the index pulse during this time.
3. If the Motor finds the Index pulse, then the Home-position is found.
4. The actual and the command position of the Drive will be set to the HOME.P value as soon as the index-signal is found. The Drive then ramps down to velocity 0 and moves back to index position.
5. If the Limit switch is active before Index signal, then the Motor changes the direction and then repeats steps 3 and 4.

Homing Mode 11 Example: Find Index Signal.

1. Select Mode 11 from the drop-down box.
2. Set the Direction to Positive.
3. In the home screen press the “set capture” button.
4. When Start is selected, the motor will move until it encounters the index reference and then it stops.
5. If Limit switch is encountered before Index reference, the Motor changes direction and searches for Index signal in the opposite direction.

Unlike Homing Mode 3, the Home position is set as soon as the index pulse is found, irrespective of the direction of motion.

Homing Mode 12: Homing to a home-switch, including mechanical stop detection

This homing mode starts a motion until a digital input, which is assigned to act as a home switch, has been activated. The home switch must be activated according to the setting of the HOME.DIR setting. The home position is found as soon as the home-switch was activated during a motion in direction of the HOME.DIR setting.

The sequence of this homing mode is as follows:

1. The motor starts to move according to the HOME.DIR setting.
2. The home position has been found as soon as the home-switch becomes active during a motion in direction of the HOME.DIR setting. The actual- and the command position of the Drive will immediately be set to the HOME.P value and the motor ramps down to velocity 0.

This homing mode is similar to the homing mode 4, but checks if the motor hits a mechanical stop instead of the hardware limit switches. A mechanical stop is detected as soon as the absolute value of the position error (PL.ERR) is larger than the position error threshold (HOME.PERRTHRESH) setting. The current command value is limited to the HOME.IPEAK value during the homing process. The motor behaves as follows when a mechanical stop has been detected before the home switch was found:

1. The motor changes the direction until the home switch is crossed.
2. The motor ramps down to velocity 0 and changes afterwards the direction again after crossing the home switch.
3. The home-switch will now be activated according to the HOME.DIR setting and the home-position has been found. The actual and the command position of the drive is immediately set to the HOME.P value and the motor ramps down to velocity 0.

If the Home Reference Input is active when told to home, the drive does a reset and then the home sequence. The sequence of the reset:

- The motor moves in the opposite direction of HOME.DIR
- When the home switch is not active, the motor ramps down to zero, and subsequently follows the sequence of the homing mode.

Homing Mode 13: Absolute Mode - Use Feedback Position

This mode should be selected when using a multi-turn feedback device with AKD. Since the feedback keeps track of its value at all times, the drive takes the value of the feedback at power up. The homing flag is also set. Use auto-home with this mode (HOME.AUTOMOVE). The multi-turn device is initially referenced using FB1.OFFSET. This value is set in the drive using the terminal screen and will need to be saved to the drive. Single-turn absolute devices can also utilize this mode if they are used in applications like a rotary index table where the entire range is within 360 degrees.

14.1.4 Using Homing: Advanced

The various homing methods in the AKD offer several options for setting up your home reference. When using any of the methods that use homing switches and limits, please refer to the Input/Output section for proper wiring techniques.

Related Parameters and Commands

HOME Parameters (pg 662)

PL.FB (pg 817)

CAP0.MODE, CAP1.MODE (pg 437): Sets index capture method

Related Topics

Digital Inputs and Outputs (pg 100)

DIN Parameters (pg 454)

DOUT Parameters (pg 472)

14.2 Motion Tasks

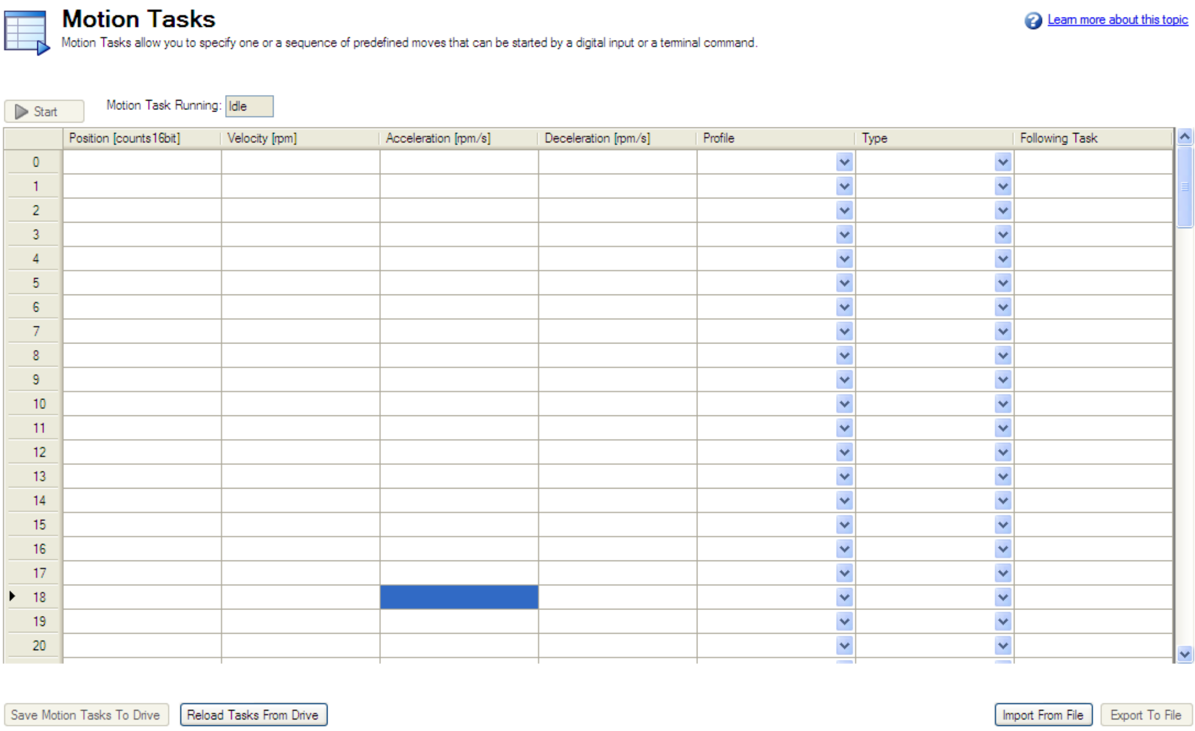
14.2.1 Overview

The AKD offers several options for executing moves, which are called "motion tasks" in WorkBench. You can directly link an input to a single motion task, assign inputs as a BCD pointer and execute, command over the Ethernet connection, or automatically execute a motion task as the result of the completion of another move event. Using the Motion Task view, you can also set up a single motion task or a sequence of motion tasks through the Motion Task screen, which is accessed from the Settings view. The Motion Task view allows you to modify a variety of parameters for a given move, including move profile, move type, velocity, distance, and acceleration limits.

NOTE Motion Tasking is not supported by AKD SynqNet.

14.2.2 Motion Task Input Table

Motion tasks can be easily entered and manipulated using the Motion Task Table. With this table, you can enter specific motion tasks and edit tasks, as well as insert and delete tasks, much like an Excel spreadsheet. The data table you build remains in WorkBench until you load the tasks into the drive. Once the tasks are loaded, you have access to the graphical representation of the moves (as in previous versions of WorkBench).

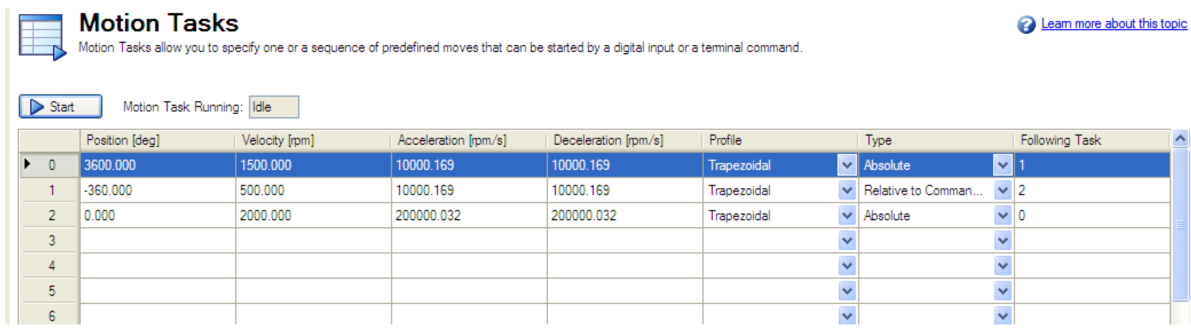


Enter values in each column for each task that you require. Once you enter an item, that field will change color indicating it has been edited but not yet loaded to the drive. Continue entering your tasks row by row. If you find that you skipped a task, you can easily insert it before or after a row that you highlight, and right-click to open the menu options. You can also select a task and using the right click access the copy and paste commands.

For delays and motion blends, double click on the line you want to edit and the delay and blend options will be available to adjust. You must load any changes or modifications to the drive before they will be available for use.

14.2.3 Using Motion Tasks

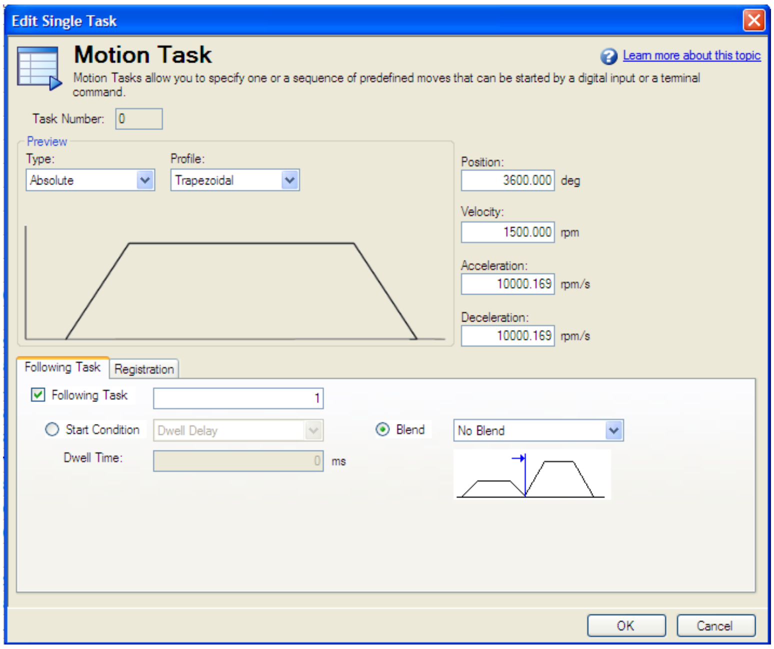
Use the Motion Task view to create and execute new motion tasks within WorkBench. As you add motion tasks, the new tasks appear as branches. You can use the root Motion Task view to view all tasks at once and execute individual tasks. When you select the Motion Task view, the Motion Task table is opened, as shown below.



From this view, you can perform the following actions:

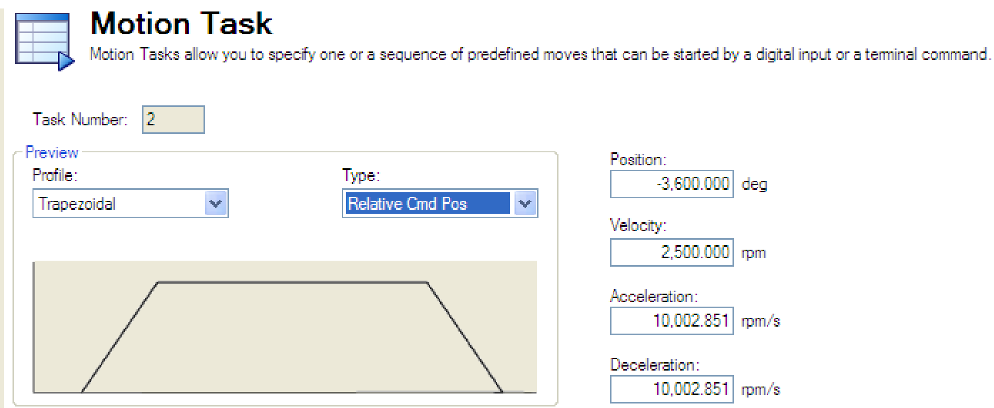
Button or Dialog Box	Description
Start	By highlighting an existing motion task, you can use this button to initiate that task (and subsequesnt tasks it may be linked to). In order to start any Motion Task, the motor must have been homed.
More/Less	This expands the Task table to include additional information regarding each task.
Save Motion Tasks To Drive	Saves the motion tasks in the table to the drive. This field is highlighted in pink if any values in the table have changed and have not been saved to the drive
Reload Tasks From Drive	Uploads motion tasks table with current values from the drive. This action deletes any values listed in the table.
Import From file	Allows import of a .xml file to the task table.
Export To File	Exports to a .xml file the current list of tasks in the table.

All of the task parameters can be added to the Motion Tasks Table when More is selected. You can also view individual tasks in the Motion Task (Edit Single Task) screen by double-clicking on the task row.



Once in the edit screen, you can adjust move type, position command, velocity and accelerations as well as sequencing options. The editable fields include:

- **Type**: Sets the type of move, absolute or incremental type moves.

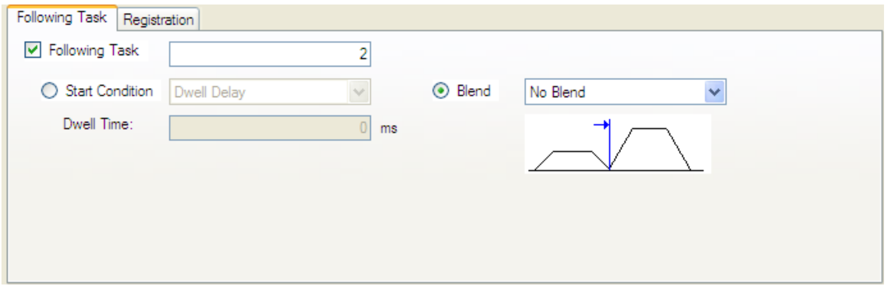


- **Profile** : This area sets the basic shape of the move. The basic move is trapezoidal, but S-curve (1:1) and custom profiles are also available using a "Profile Table".

- **Position**: This is the command position (PL.CMD), based on the type of move selected.
- **Velocity**: Sets the peak or traverse velocity depending on the move parameters
- **Acceleration**: Sets the profile acceleration ramp (this cannot be set higher then the drive acceleration limit setting - DRV.ACC)
- **Deceleration**: Sets the profile deceleration ramp (this cannot be set higher then the drive deceleration limit setting - DRV.DEC)

In addition to these profile settings, you can set additional parameters on how your next move will be executed. These parameters include:

- **Following Task**: By checking this box, you can indicate which task you want to follow the task that you are editing.



- **Start Condition**: This box allows for different start conditions to initiate the following task. Currently, the only option is a dwell delay. The dwell delay time can be entered in the dwell time box.
- **Blend**: It is possible to blend moves together by using the blend option. This option allows either a blend to velocity, or blend to acceleration.

Once you have entered the data, you can click on the Motion Task root directory to bring the table up to execute your moves. For more advanced motion tasks, specific constraints can be added. Currently, you can have a task that is either an interrupt task, or non-interruptible.

Advanced: Constraints

- **Interrupt**: This constraint can be used to interrupt a task that has not been finished. In other words, this constraint only will initiate the interrupt task motion if another task with no constraints is underway. This selection is a good option for registration tasks when you only want the move to happen if the current task has not finished.
- **Non-Interruptible**: When this constraint is selected, then the task can not be interrupted by another motion task or task call from inputs.

14.2.4 Motion Profiles

Trapezoidal moves include a fixed ramped acceleration (set by MT.ACC), a traverse period at velocity (set by MT.VEL), and a fixed ramped deceleration (set by MT.DEC). The distance traveled during the move is determined by the type of move (absolute or incremental). Position is set using MT.P. Note that for short moves, the traverse velocity may not ever be reached.

14.2.5 Motion Types

Motion tasks can use the following motion types:

- Absolute
- Relative to command position (PL.CMD)
- Relative to previous target position
- Relative to feedback position

Motion types define how the target position will be calculated. These can be either incremental (relative) or absolute. An incremental move is a move that increments a specific distance. The AKD allows the user to increment based on two different beginning positions as described below. Absolute moves move to a specific position based the actual position form an established zero or home point.

Absolute motion task

An absolute move type will move to the actual position indicated by MT.P. A home move would have been established to provide the reference for actual positions on the machine. In this case, the target position = PL.CMD=MT.P

Example:

You want to move to a position that is 68 degrees from home reference.

On the Motion Task edit page, Select the Absolute move type, and then enter 68 into the position block (units should be set in degrees). Enter the traverse speed and adjust the acceleration and deceleration parameters as needed. Once you move off the edit screen you can select this task and initiate a start. Regardless of your current position, the motor will now rotate to the absolute position of 68 degrees as referenced from the home position.

Motion task relative to command position (PL.CMD)

This type is simply an incremental move. The target position is based on the current position represented by PL.CMD plus the increment you want to move. Specifically, Target Position = PL.CMD + MT.P.

Example:

Your current motor position at the time you activate the motion task is 38 degrees. You want to move an increment of 30 degrees.

On the Motion Task edit page, select the **Relative to Command Position** move type, and then enter 30 into the position block (units should be set in degrees). Enter the traverse speed and adjust the acceleration and deceleration parameters as needed. Once you move off the edit screen you can select this task and initiate a start. The motor will move 30 degrees form the current position. After the motion task, the motor will now be at 68 degrees (38 + 30 = 68).

Motion task relative to previous target position

Using the last target position as the start point, this profile will move the increment chosen from that old position. This motion type is recommended in situations where a previous task may have been interrupted, or you want to eliminate any accumulated error. The target position will look at the previous target position and then add the increment you enter for this task. Specifically, Target Position = Previous Target Position + MT.P.

Example:

You initiated a motion task to increment 360 degrees, but that task was interrupted and the motor was stopped at 175 degrees. You use this method and increment another 360 degrees, the motor will complete the motion at 720 degrees (basically, it finished the first move to 360, and then made the additional distance requested of 360 degrees).

The task would be set up similar to the Relative command position example above. To view this in action, set up 2 tasks, once Relative Command Position and the second, Relative Old Target Position. Use low velocities so you can stop motion before the move is completed. Start the first move and then stop before it finishes. Then select the second move. The motor will stop at the desired end position if the stop had not occured. Try this again, but do not use the Relative Old Target Position and you will see the difference.

14.2.6 Using Motion Tasks: Advanced

Joining multiple tasks

Tasks can be joined in sequence using the task setup screen. Dwells can be added to allow the following tasks to wait a specific amount of time before starting. Tasks can also be blended to provide for smoother transitions between multiple moves. AKD allows blending with either acceleration or velocity.

Start Conditions

AKD currently provides only one alternatives to start a following task at this time (more are planned):

- Dwell Delay. Starts the following task after the dwell time entered (MT.FTIME)

Blending

As an alternative to dwells, the AKD can blend motion tasks to provide smooth transitions between multiple tasks. This feature extends the life of your machine life by minimizing mechanical wear due to hard move transitions (jerks).

- **No blend**. No blend, move starts upon completion of the previous task
- Blend into Acceleration. Blends the acceleration of the current task into the following task. As the target position of the first task is reached, the acceleration blends into the second task. This prevents the motor from decelerating to zero before starting the second move. This only works where both accelerations are driving the motor in the same direction.
- Blend into Velocity. Blends the velocity of the current task into the following task. In this method, the target position is reached as the velocity of the second move is reached. The blending begins prior to reaching the target position, and is completed at the target position of the first move, and the traverse velocity of the second move. This only works when both velocities are in the same direction.

Related Parameters and Commands

MT Parameters and Commands (pg 779)

DRV.MOTIONSTAT (pg 541): Bit 0 monitors if motion tasking is active.

AIN Parameters (pg 364)

AOUT Parameters (pg 393)

DIN Parameters (pg 454)

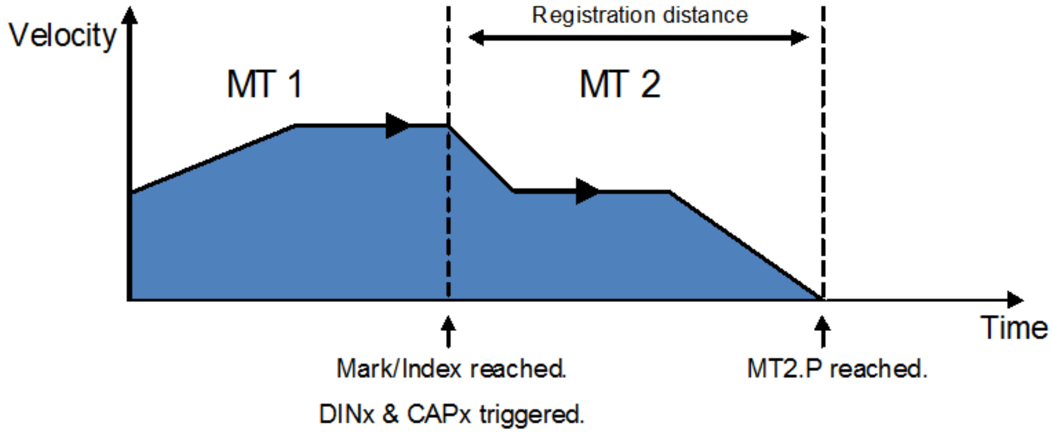
DOUT Parameters (pg 472)

HOME Parameters (pg 662)

14.2.7 Registration Moves

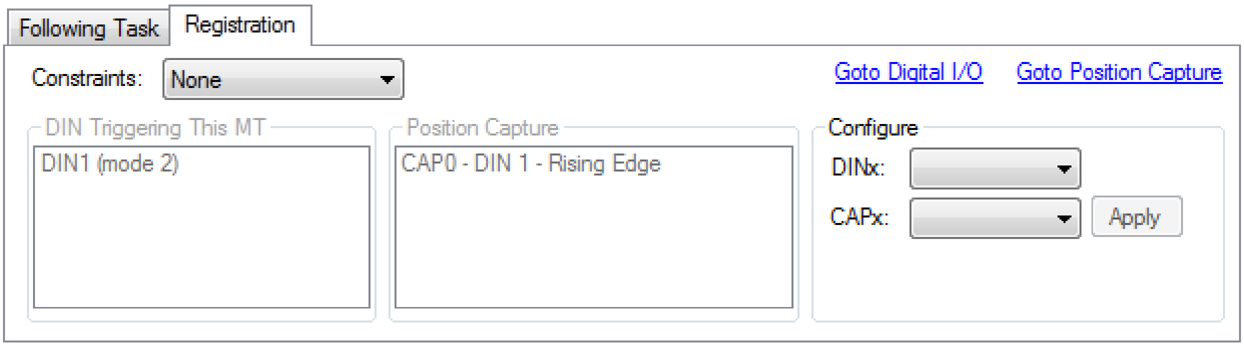
Registration moves are also known as "indexing on the fly." In a registration move, a digital input interrupts a running motion task and starts a new one. The start position of the new motion task is latched at the time the digitial input is activated. The target position of the new motion task is calculated based on the latched position value, for a very accurate target position.

Typical applications for using the registration move are feed-to-length applications, which must guarantee proper positioning with respect to special mark or index. If this mark is reached, an external trigger signal aborts the current move and starts the registration move.



14.2.7.1 Configuring Registration Moves in WorkBench

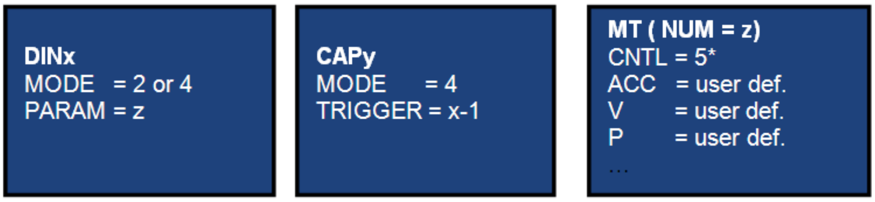
You can configure registration moves from the Motion Task editor:



Box	Description
Constraints	Configures bits 13 and 14 from the MT.CNTL keyword. For registration "none" or "interruptable" should be used. "Non-interruptable" will not function properly. See for information on constraints.
DIN Triggering (read-only)	Displays the list of all digital inputs that are configured to start motion tasks. These are potentially all digital inputs that can trigger this motion task. This field is read-only, indicating current configuration.
Position Capture (read-only)	Displays the list of capture engine that are configured with the proper Capture Mode (4-auto-armed position) in order to execute a registration move. These are potentially all registered position that can be used for this MT.
Configure (write-only)	Configures a digital input and the capture engine to be use as registration source for this MT.

14.2.7.2 Configuring Registration Moves from the Terminal View

You can also configure registration moves from the terminal view using drive parameters. To configure a registration move, you must configure three sets of parameters.



*other options are possible (Bit 13 & 14); see Registration Moves (pg 162) and the MT.CNTL (pg 783) parameter description.

- Digital input (DINx) : Configure DINx to mode 2 or 4 so that input x will trigger the registration move.
- Capture (CAPy): Either capture engine 0 or 1 may be used to trigger a registration move. Set CAPy.MODE to 4, where "x" indicates the capture engine to be used. CAPy.TRIGGER must be set to x-1, where x is the DIN used above. All other capture parameters can be set as desired (see capture parameter descriptions).
- Motion tasking (MT): A registration move requires the standard motion task parameters (ACC, DEC, V, P). It also requires that the motion control word be set as follows:

  0x0005 - standard registration move (this move can be interrupted and can start regardless of previous velocity)

  0x2005 - non-interruptible registration move (move must complete before any other motion task can start)

  0x4005 - interruptible registration move but will not start if velocity is 0

  0x6005 - non-interrutible registration move that will not start if velocity is 0

  After the motion task is configured MT.SET can be used to complete the setup of the motion task.

Example

Mark triggers digital input 2.

Capture machine 0 is used.

Motion task 3 is the registration move.

Motion task 3 is only activated if a preceeding motion task is active.

Motion task 3 is configured with 1,000 rpm/s^2 acceleration and deceleration , target velocity of 10 rpm and a relative position of 50,000 counts.

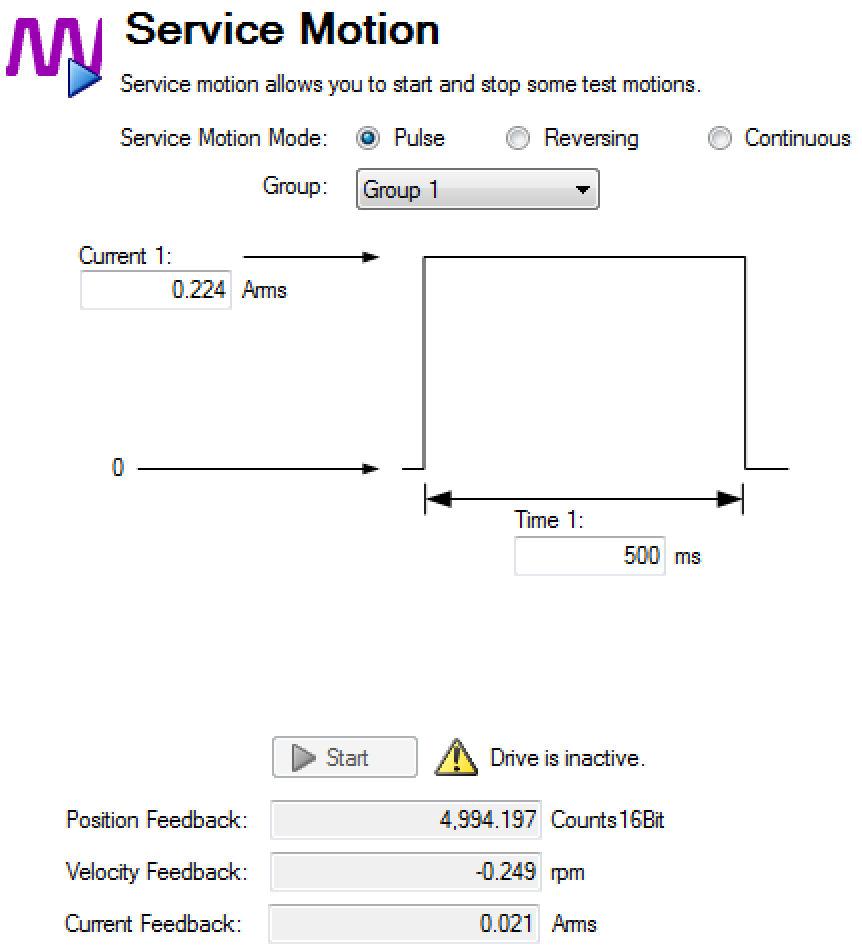
Commands:

```
DIN2.MODE 2
DIN2.PARAM 3
CAP0.MODE 4
CAP0.TRIGGER 1
MT.NUM 3
MT.CNTL 16389
MT.P 50000
MT.ACC 1000
MT.DEC 1000
MT.V 10
MT.SET
```

The registration move is now active.

14.3 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 repetative 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.



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

Button or Dialog Box	Description
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.
Reversing	Alternates between the two commanded values. You can specify the time the command will be held in each state.
Continuous	Runs the commanded current or velocity continuously.
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 (pg 881) for more details.
Current 1/Current 2	Sets the two different currents.
Time 1/Time 2	Sets the time for which the different commands are generated. Setting a time to zero generates a continuous command.
Start/Stop	Starts and stops the motion.
Position Feedback	Displays the present position of the motor.

Button or Dialog Box	Description
Velocity Feedback	Displays the present velocity of the motor.
Current Feedback	Displays the present current of the motor.

The **Drive Motion Status** view indicates when the drive is performing service motion.

Related Topics

Drive Motion Status (pg 176)

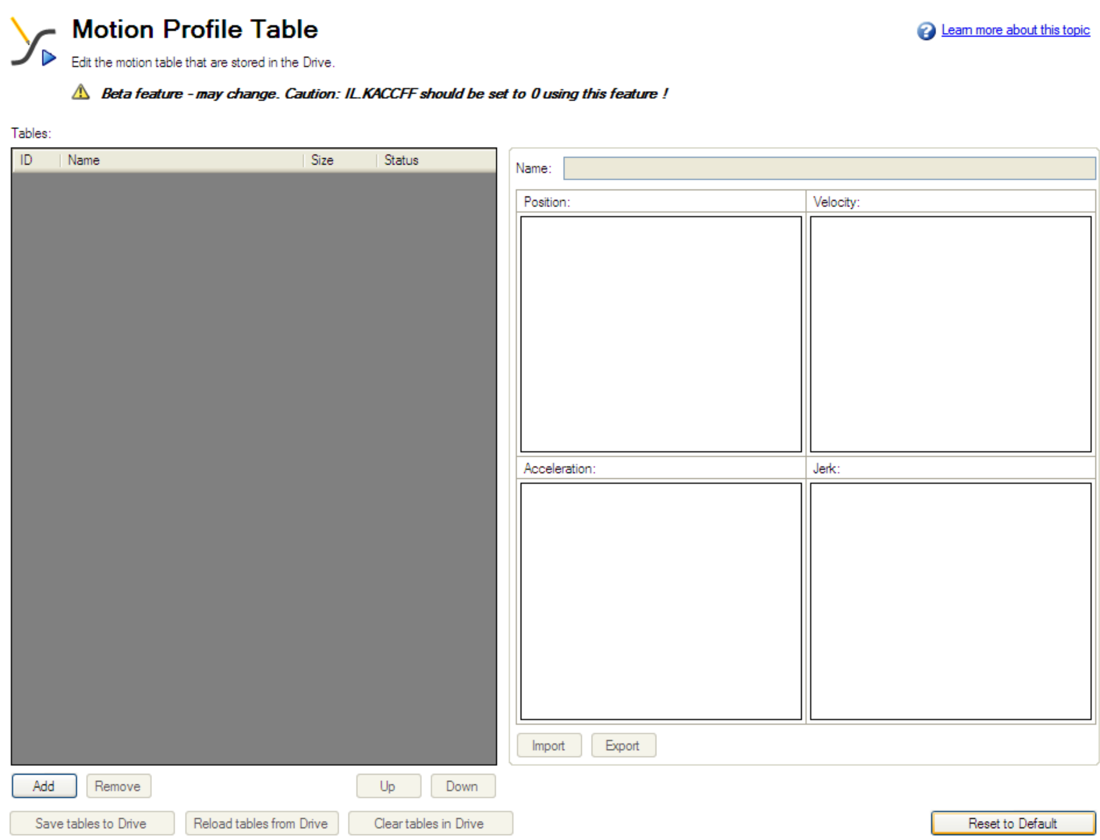
Related Parameters

SM.I1 (pg 879)
SM.I2 (pg 880)
SM.I2 (pg 880)
SM.MOVE (pg 884)
SM.T1 (pg 885)
SM.T2 (pg 886)
SM.V1 (pg 887)
SM.V2 (pg 889)

14.4 Motion Profile Table

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.



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

- Grid
- Graphical Representation
- Control Buttons

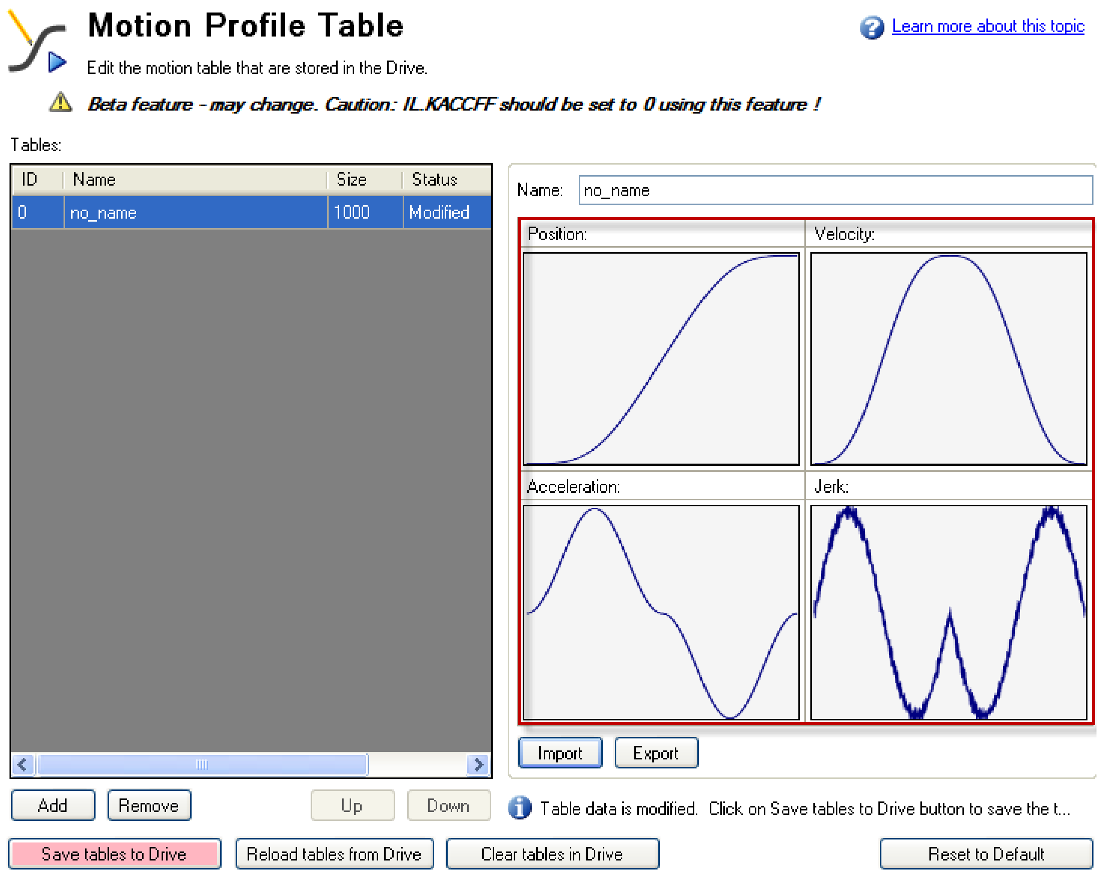
14.4.1 Grid

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".

14.4.2 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:



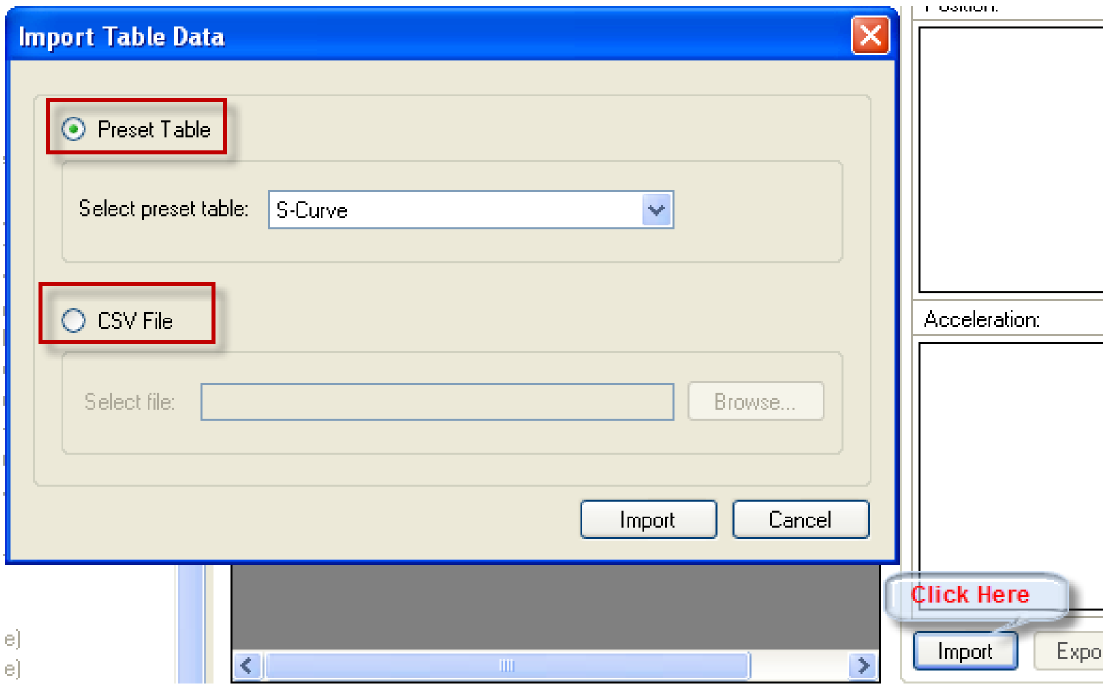
14.4.3 Control Buttons

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.
Up/Down	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.
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.

14.4.3.1 Import Table Data

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



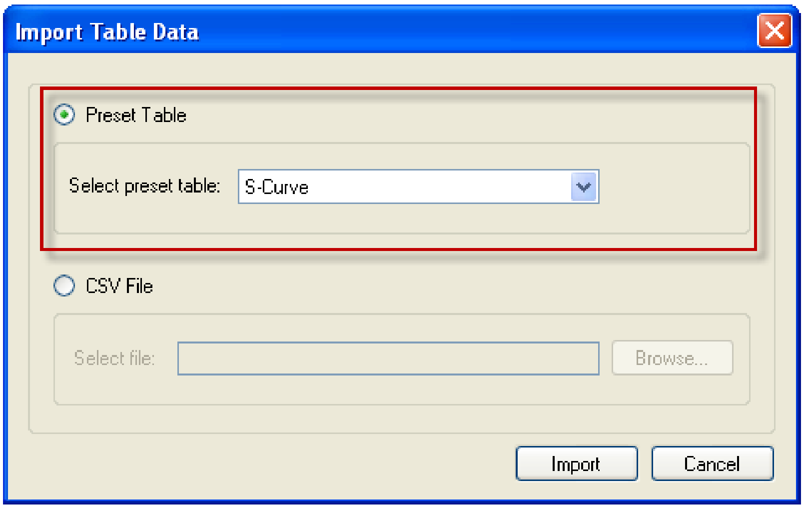
You have two options to provide inputs to the profile table:

- Preset Table
- CSV File

14.4.3.2 Importing data from Preset Table option

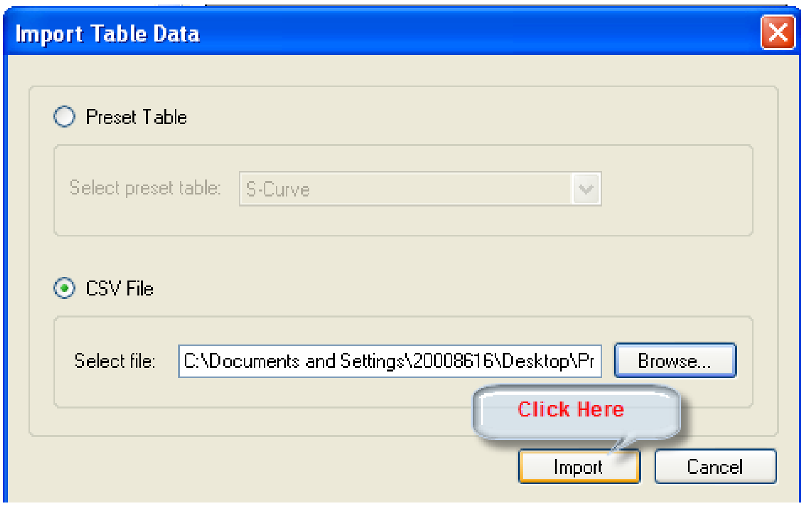
By default, the **Preset table** option is selected. You can select one of the preset tables from the **Select Preset Table**box. After selecting the source, click **Import**to complete the import of the data.

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.



14.4.3.3 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.

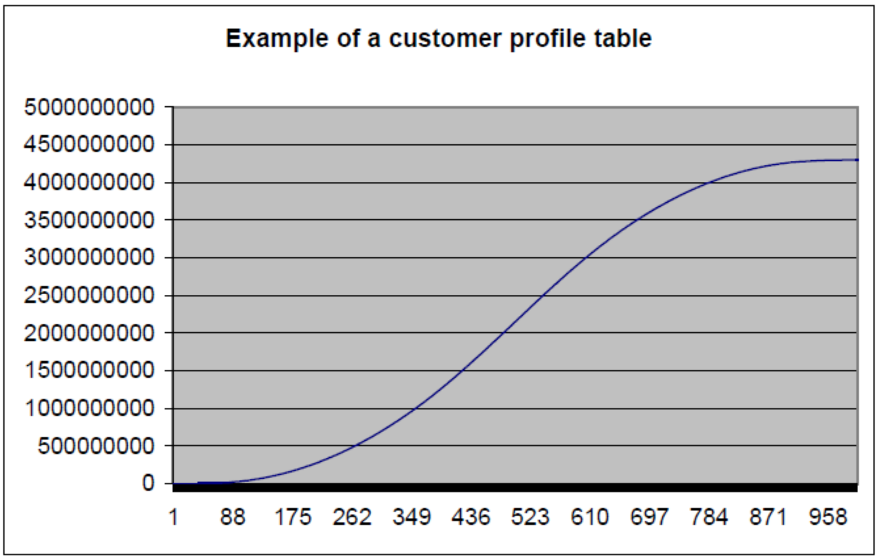
14.4.4 Motion Profile Table: Advanced

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.

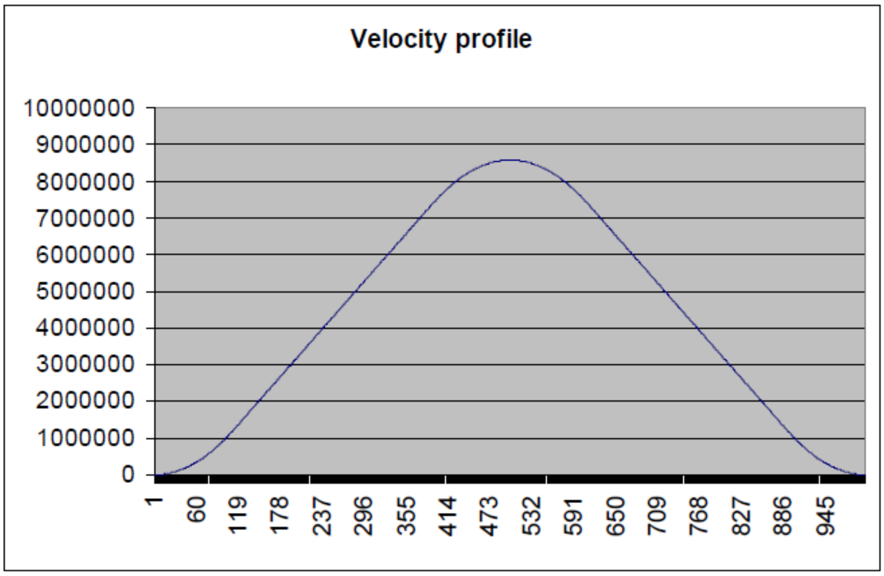
The motion profile table describes the shape of the acceleration process, but does not determine how fast the motion task accelerates or decelerates and which target velocity will be reached.

14.4.4.1 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:

$velocity\_profile\_value_n = customer\_profile\_entry_{n+1} - customer\_profile\_entry_n$

14.4.4.2 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.
4. The following motion profile table entry must contain the value of nearly 2^{31}.

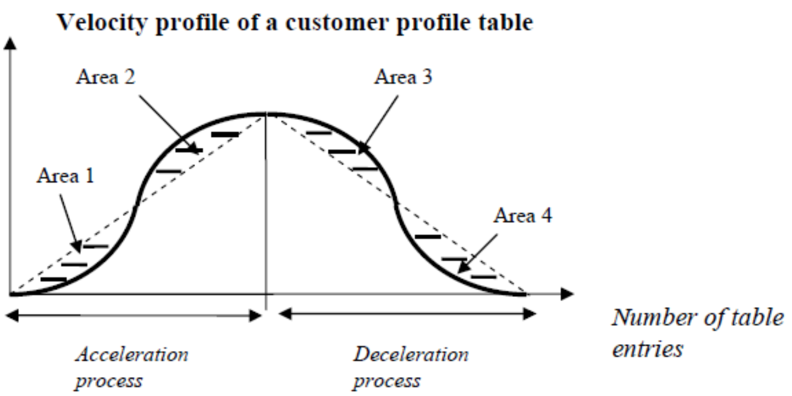
$$table\_entry = \frac{number\_of\_table\_points}{2} + 1$$

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.

5. A motion profile table must also be symmetric during the acceleration and the deceleration process when a standard customer table motion task must be triggered.

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

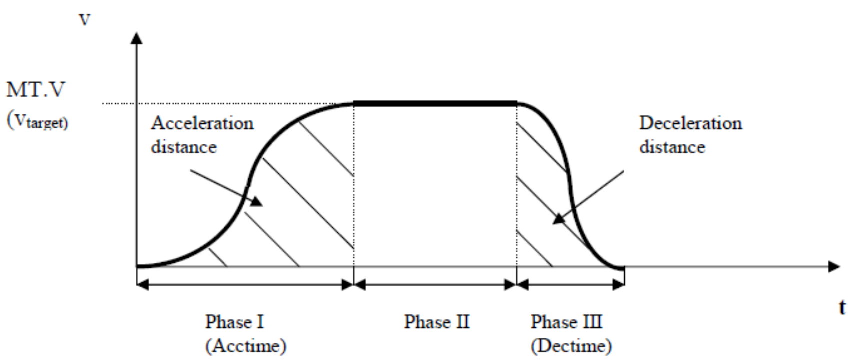


The left half of the curve describes the shape of the acceleration process of the motion task. The right half of the curve describes the shape of the deceleration process of the motion task. A symmetric motion profile table means that Area 1, Area 2, Area 3 and Area 4 have the same size.

14.4.4.3 Different methods of motion table motion tasking

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 1:1 customer table motion task. The diagram below illustrates a basic table profile algorithm. The figure shows a standard customer table motion task.

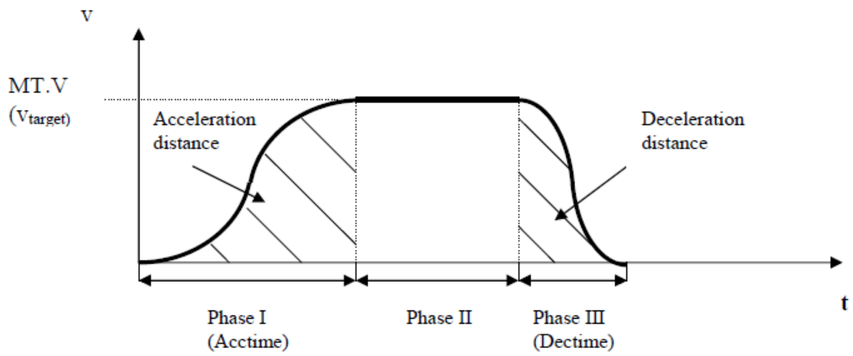


The drive calculates the acceleration time and deceleration time out of the given motion task parameters (see MT Parameters and Commands (pg 779)) with the assumption of a trapezoidal acceleration setting (MT.ACC (pg 780) and MT.DEC (pg 787)). The formulas are:

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

14.4.4.4 Standard customer table motion task

The standard motion profile table motion task is displayed in the following figure:



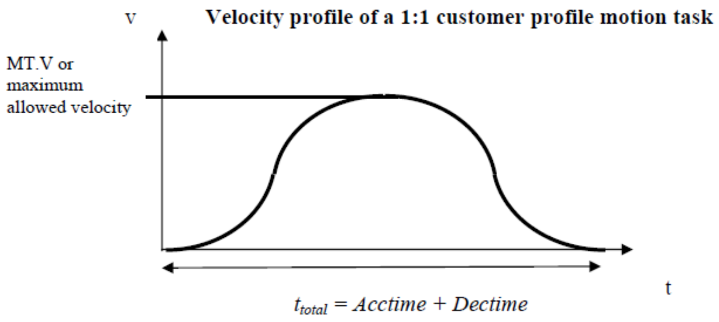
The standard handling for a stand-alone motion task, which means that this motion task does not trigger automatically a following motion task, can be separated in three different phases:

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 the chapter Restrictions for a customer table.

14.4.4.5 1:1 customer table motion task

The 1:1 customer table motion task is basically very similar to the standard customer table motion task handling with just a few small differences.

1. The 1:1 customer table motion task does not step out of the table after an acceleration process and inserts a constant profile (Phase II in the chapter above). The 1:1 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 1:1 profile does not use different acceleration and deceleration values. The AKD calculates the sum of the acceleration time and deceleration time and uses this total time (total¬= 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 (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 (see chapter 1.2 for more details). The velocity will be different in case that the customer table is non-symmetric.

14.4.4.6 Setting up a motion profile motion task

It is recommended to set-up any motion task via the AKD Workbench PC software. The S-curve profile and the 1:1 profile will 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.xyz commands. There are 2 statements which are mentioned within this chapter:

- A trapezoidal acceleration, a 1:1 customer table motion task or a standard S-curve motion task will be selected via the bits 10 and 11 of the MT.CNTL command.
- The MT.TNUM parameter describes for each motion task, which table to use for the 1:1 customer table motion task or the standard S-curve motion task. The parameter MT.TNUM will be ignored in case that a trapezoidal motion task has been selected.

For more details please refer to the AKD Parameter and Command Reference documentation in the AKD Workbench help menu.

14.4.4.7 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.

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.

1:1 customer table motion task

As already described in chapter 2.3, it is not allowed activating a 1:1 profile motion task while another motion-task is currently running. A 1:1 customer table motion task must start from velocity 0.

When activating a 1:1 customer table motion task the AKD pre-calculates the expected peak-velocity and check is the velocity exceeds the minimum of the MT.V, VL.LIMITP and VL.LIMITN limitation.

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

$$v_{PeakExpected} = \frac{2 \cdot Distance\_to\_travel}{t_{total}}$$

The ‘distance to travel’ is defined in the motion task settings MT.P & MT.CNTL. In case that VPeakExpected exceeds the minimum of the MT.V, VL.LIMITP or VL.LIMITN setting, the AKD re-calculates ttotal in a way, that VPeakExpected does not exceed the velocity limitations.

The AKD accelerates and decelerates within the same time in case of a 1:1 profile and therefore different settings for MT.ACC and MT.DEC are not considered.

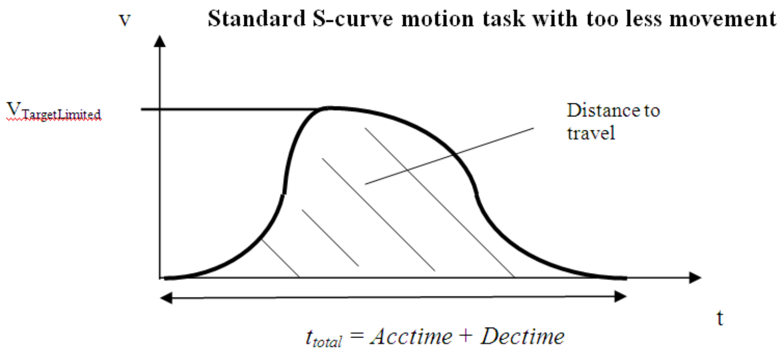
14.4.4.8 Standard customer table motion task

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

Similar to the considerations in chapter 4.1 the 'distance to travel' of a motion task is specified by the motion task settings MT.P & MT.CNTL. Furthermore the target velocity of the motion task (MT.V) and the acceleration and deceleration (MT.ACC and MT.DEC) are part of the motion task settings.

An impossible motion task setting would be, 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). In this case the AKD lowers the target velocity automatically to VTargetLimited and accelerates within the selected acceleration time to the limited target velocity and decelerates afterwards 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.



During a change on the fly condition

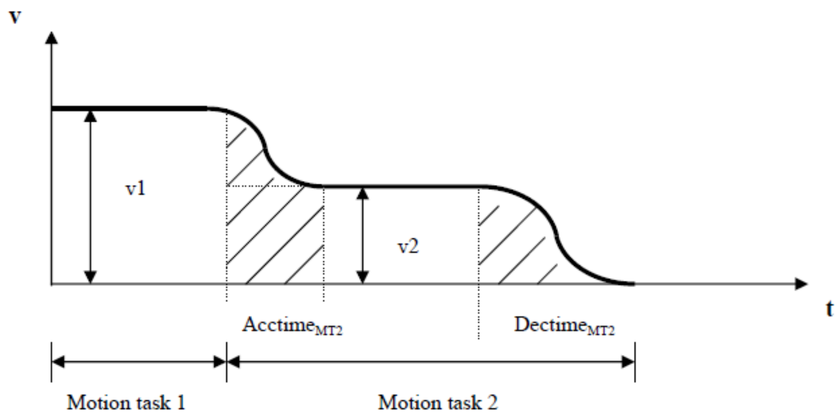
There are 2 different kinds of considerations within the AKD firmware for a change-on-the-fly condition.

- 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).

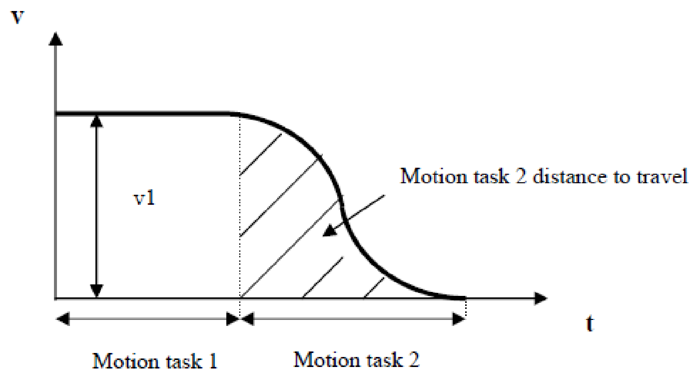
Since the shape of a customer table is unknown to the AKD, the Drive verifies in advance the validity of the motion task with the assumption of a symmetric motion profile table.

Movement to the same direction

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

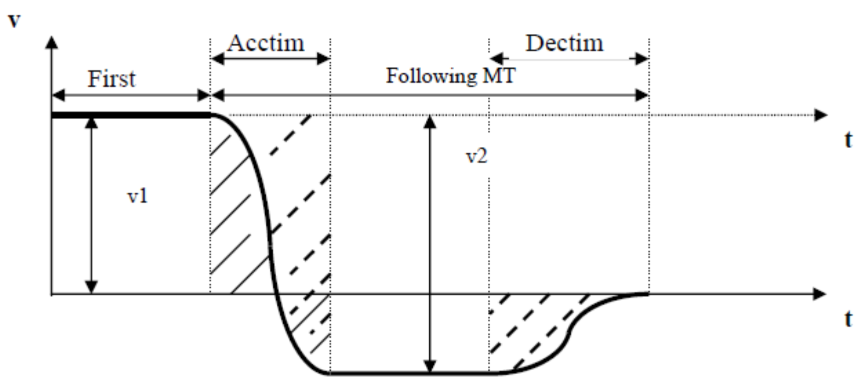


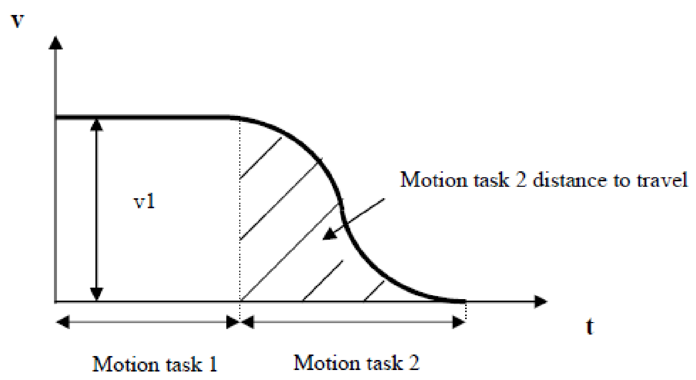
In case that the distance to the target position of the motion task 2 is smaller than distmin, the AKD generates a profile as shown in the next figure.



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 v1 to v2. 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 v1 to v2 also a bit in positive direction. The magnitude of v2 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 drives 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.

14.5 Jog Move

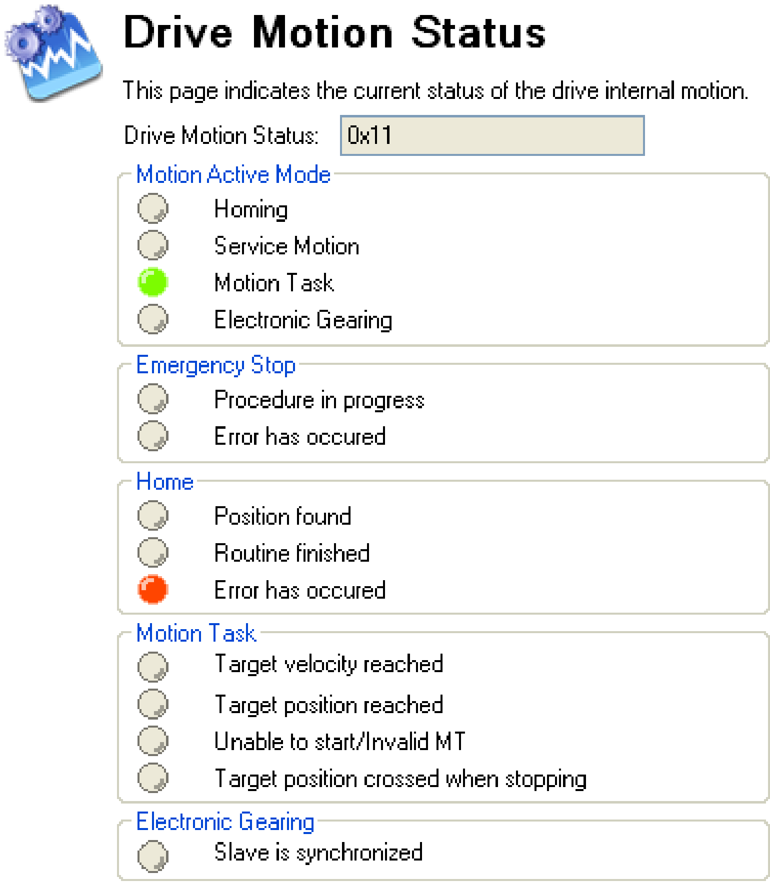
This screen verifies that the servo system is able to command motion. Verify the settings for velocity, acceleration, and deceleration. Adjust these settings if necessary. Click **Jog** and continuous motion will occur until you click **Stop**. If motion does not occur, check for warnings, faults, or prompts from WorkBench.

If erratic motion and or vibration occur when motion is commanded, open the wizard **Tuning** screen and reduce the desired bandwidth significantly. If erratic motion continues, exit the **Setup Wizard** and use the **Performance Servo Tuner** and **Service Motion** screens in the settings tree for more advanced setup of the servo system.

NOTE In order to tune the system the drive must be in service-type command source and either velocity or position opmode. If the drive is in torque opmode, then a popup screen will allow you to switch into velocity mode.

14.6 Drive Motion Status

Drive Motion Status allows you to view the current status of the drive internal motion. An LED indicates the status of various possible motion states (as read by the DRV.MOTIONSTAT (pg 541) parameter). The **Drive Motion Status** box displays the DRV.MOTIONSTAT (pg 541) output in hexadecimal form. The boxes below **Drive Motion Status** indicate the drive state. When drive motion is activated, a green LED is shown. When error occurs, a red LED is shown as indicated below:



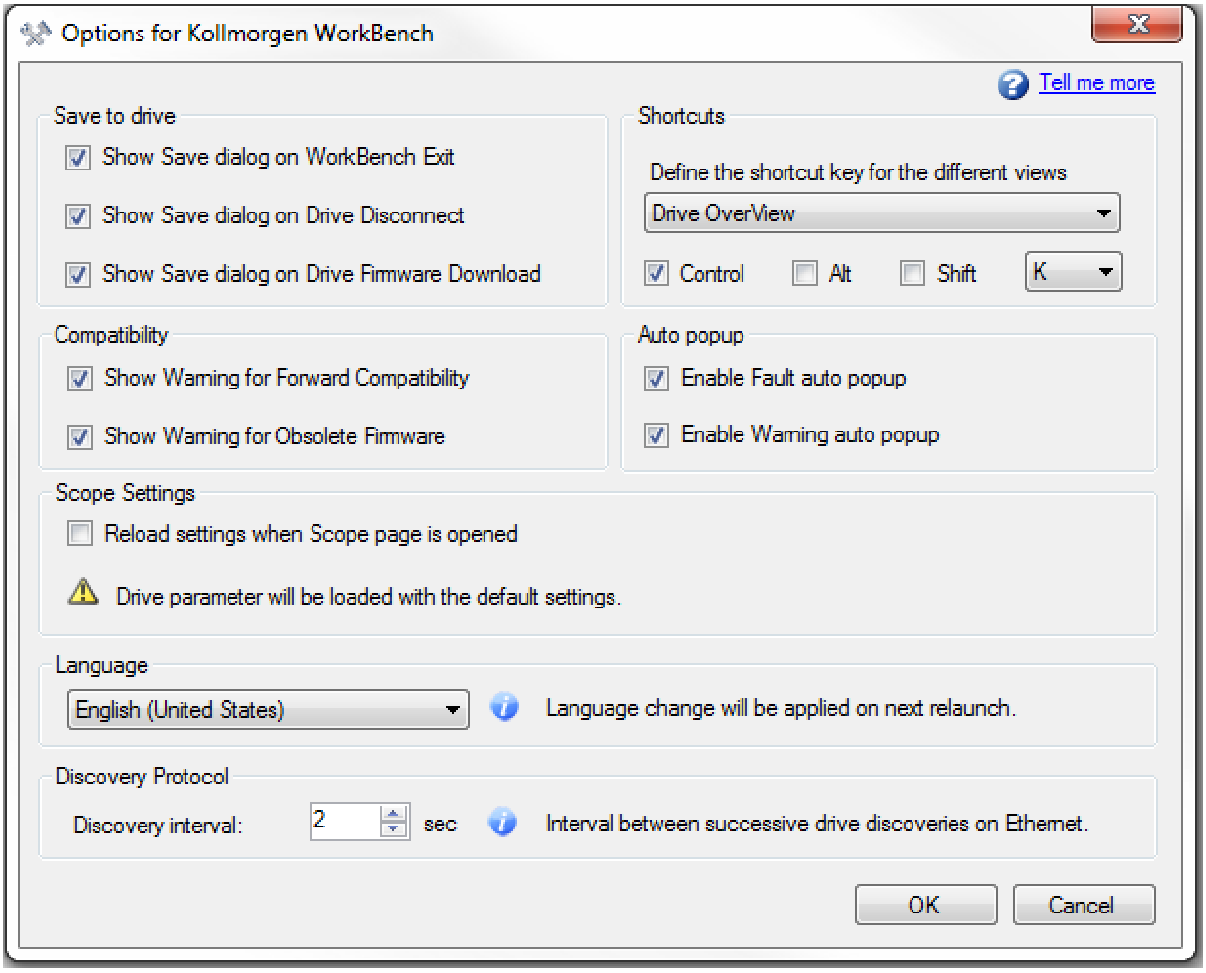
Related Topics

Service Motion (pg 164) | Emergency Stop (pg 128) | DRV.MOTIONSTAT (pg 541)

15 Saving Your Drive Configuration

15.1 Save Options

WorkBench offers several options for saving your drive configuration. From the WorkBench menu, if you select **Tools** and then **Options**, the following window appears:



The **Save** options are applied as follows:

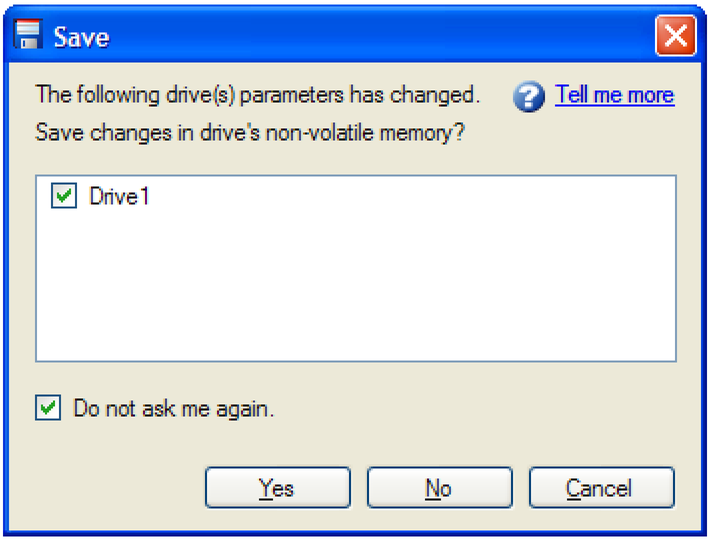
Button or Dialog Box	Description
Show Save dialog on WorkBench exit	If this box is checked, then every time you exit WorkBench a dialog will ask you if you wish to save the drive parameters to nonvolatile memory. If this box is not checked, the dialog will not be shown.
Show Save dialog on Drive Dis-connect	If this box is checked, then every time you disconnect from a drive a dialog will ask you if you wish to save the drive's parameters to nonvolatile memory. If this box is not checked the dialog will not be shown.
Show Save dialog on Drive Firmware Download	If this box is checked and any parameter is changed, then every time you down-load a firmware to a drive, a dialog will ask you if you wish to save the drive’s parameters to nonvolatile memory. If this box is not checked, the dialog will not be shown.

When the **Scope Settings** box is checked the default drive settings will be overwritten with selected Scope setting parameters each time the Scope page is opened. Otherwise, the drive will with default drive settings.

WorkBench is available in German and English. After making a selection from the **Language** drop down menu, click OK, and exit WorkBench. The next time WorkBench is started it will load with the selected language.

15.2 Save On Exit

When you exit WorkBench while connected to a drive, the following dialog box appears:



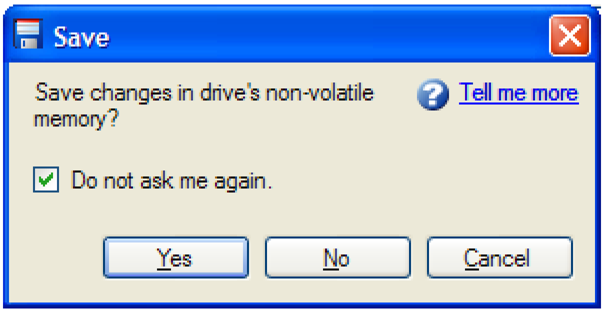
As you are working with a drive, all the changes you make are stored in the volatile memory within the drive. If you power cycle the drive or loose power to the drive then any changes you have made to the drive would be lost. You can save the drive parameters to nonvolatile memory at any time and these saved parameters will be restored when the drive next powers up.

While you are connected to the drive, WorkBench monitors all the changes you make to the drives parameters. An asterisk in the navigation tree shows if a parameter has been changed. If you do not change any drive parameters, this dialog will not be shown.

Button or Dialog Box	Description
Yes	Saves the parameters to the nonvolatile memory within the selected drives and then exits WorkBench.
No	WorkBench will exit. No drive parameters will not be saved to nonvolatile memory.
Cancel	This will stop the exit command and WorkBench will remain open.
Do not ask me again	If you check this WorkBench will not show this dialog again. There is an option in the Options dialogue to restore this setting.

15.3 Save On Disconnect

When you disconnect from the drive, you may see this dialog box:



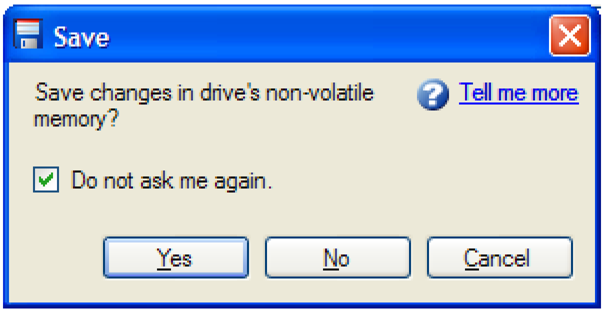
As you work with a drive, all the changes you make are stored in the volatile memory within the drive. If you restart the drive or lose power to the drive, then any changes you made to the drive would be lost. You can save the drive parameters to nonvolatile memory at any time, and these saved parameters will be restored when the drive next powers up.

While you are connected to the drive, WorkBench monitors all the changes you make to the drive parameters. An asterisk in the navigation tree shows if a parameter has been changed. If you do not change any drive parameters, this dialog box will not be shown.

Button or Dialog Box	Description
Yes	Saves the parameters to the nonvolatile memory within the drive and then finish disconnecting.
No	Disconnects the drive. The drive parameters are not saved to nonvolatile memory.
Cancel	Stops the disconnect command. You remain connected to the drive.
Do not ask me again	Does not diplay this dialog box again. If you check this box, WorkBench does not show this dialog again. The **Options** dialog includes a command to restore this setting.

15.4 Save On Firmware Download

When you download firmware to the drive, you may see this dialog box:



As you work with a drive, all the changes you make are stored in the volatile memory within the drive. If you restart the drive or lose power to the drive, then any changes you made to the drive are lost. To save these changes, you can save the drive parameters to nonvolatile memory at any time. If you save the changes to the nonvolatile memory, these saved changes will be restored when the drive next powers up.

While you are connected to the drive, WorkBench monitors all of the changes that you make to the drive parameters. An asterisk in the navigation tree indicates that a parameter has been changed. If you do not change any drive parameters, this dialog box will not be shown.

Button or Dialog Box	Description
Yes	Saves the parameters to the nonvolatile memory within the drive and then opens dialog box for the user to select the firmware file for downloading.
No	The drive parameters are not saved to nonvolatile memory. It opens dialog box for the user to select the firmware file for downloading.
Cancel	Stops the download command.
Do not ask me again	Does not display this dialog box again. If you check this box, WorkBench does not show this dialog again. The Options dialog includes a command to restore this setting.

16 Tuning Your System

16.1 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 WorkBench.

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 Biquad 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 sytem 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.

16.2 Slider Tuning

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

16.2.1 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.

16.2.2 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.

16.2.3 Inertia Ratio

If you know the inertia ratio of your load, then entering it can improve the performance of your system. If you do not know the inertia of your load then WorkBench will assume a ratio of 1:1, which will give good performance in many configurations. The inertia ratio is the ratio of your load with respect to the inertia of your motor.

16.3 Using the Performance Servo Tuner

16.3.0.1 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. The PST can work in a “one button” mode that requires no decisions from the user. It can also be set up in specific modes to control how the PST operates for specific requirements. Finally, the PST collects frequency response data (a Bode plot) that can be used for advanced analysis.

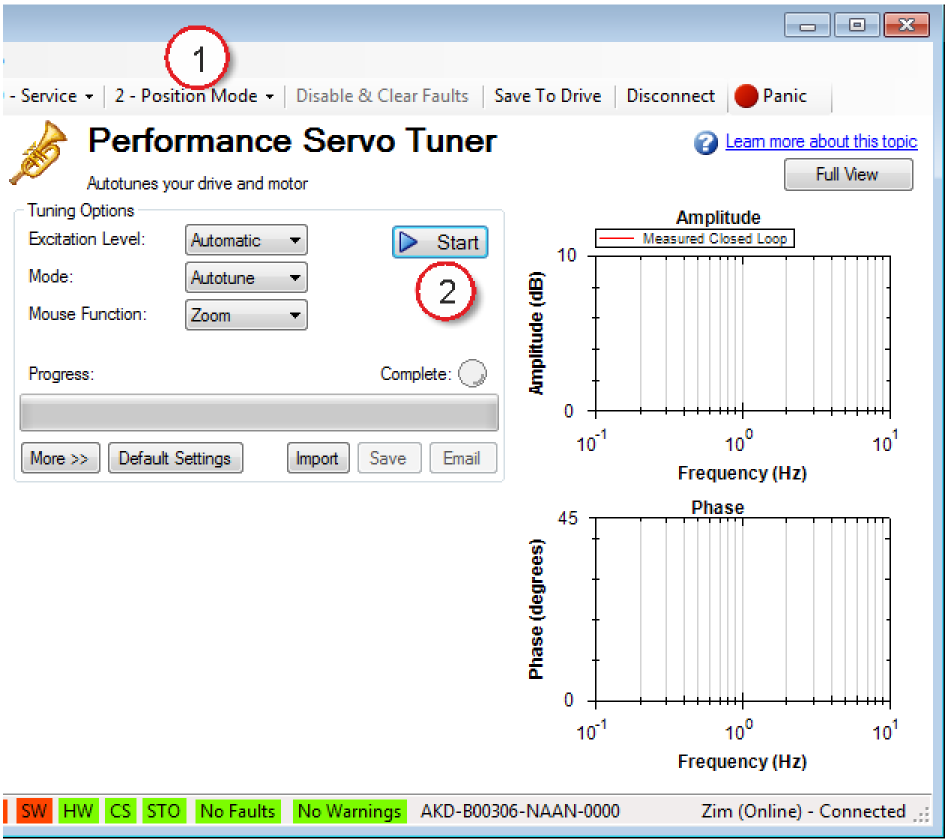
NOTE The PST will not work reliably on a vertical axis since the motor may rest against an end

NOTE stop; in this case, the inertia will not always be correctly identified.

16.3.0.2 Using the PST

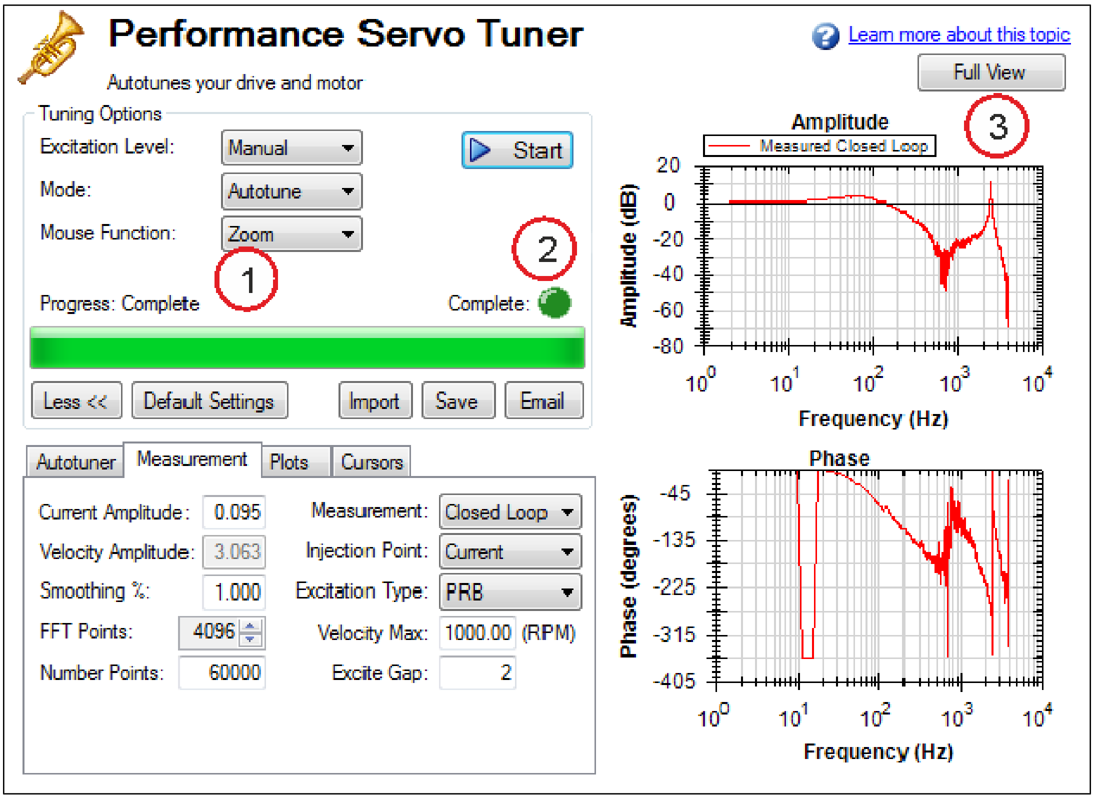
The PST defaults to a “one button” mode, in which the PST is completely automatic after pushing the start button. In the **Settings** select your desired operation mode,navigate to the **Performance Servo Tuner** view, and then tune your system as follows:

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**.



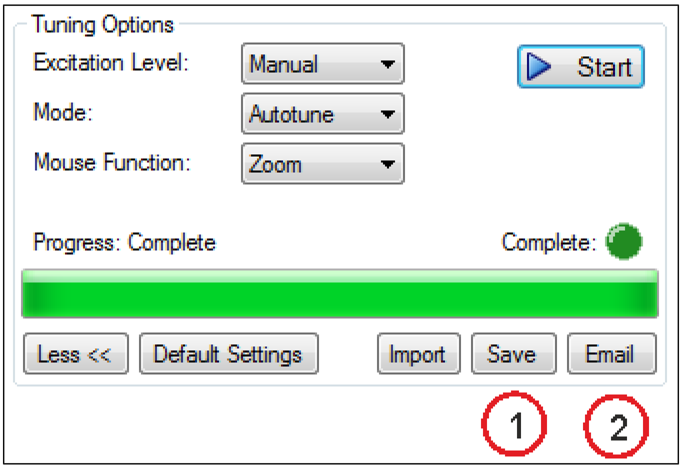
.

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.

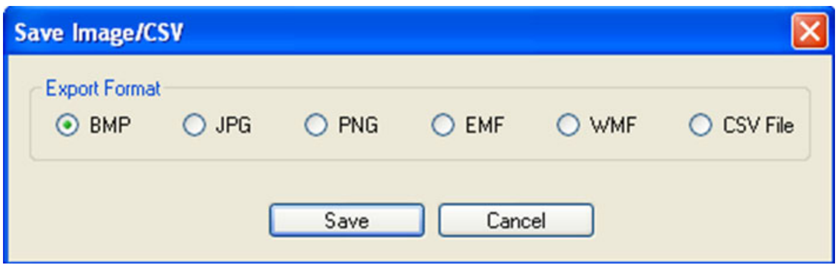


Saving and Emailing Bode Plots

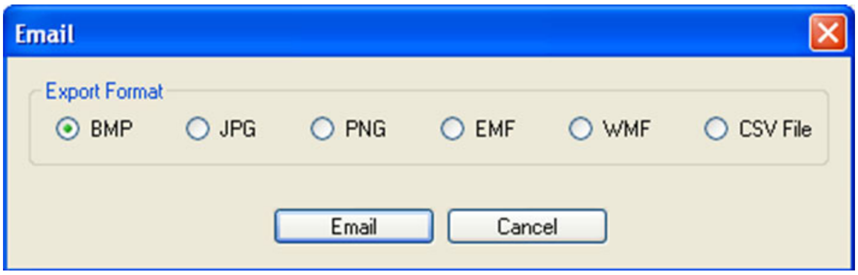
To save screenshots and raw data of a Bode plot, click on either **Save** (1), or **Email** (2).



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.



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.



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.

16.3.0.3 Measurement Options

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

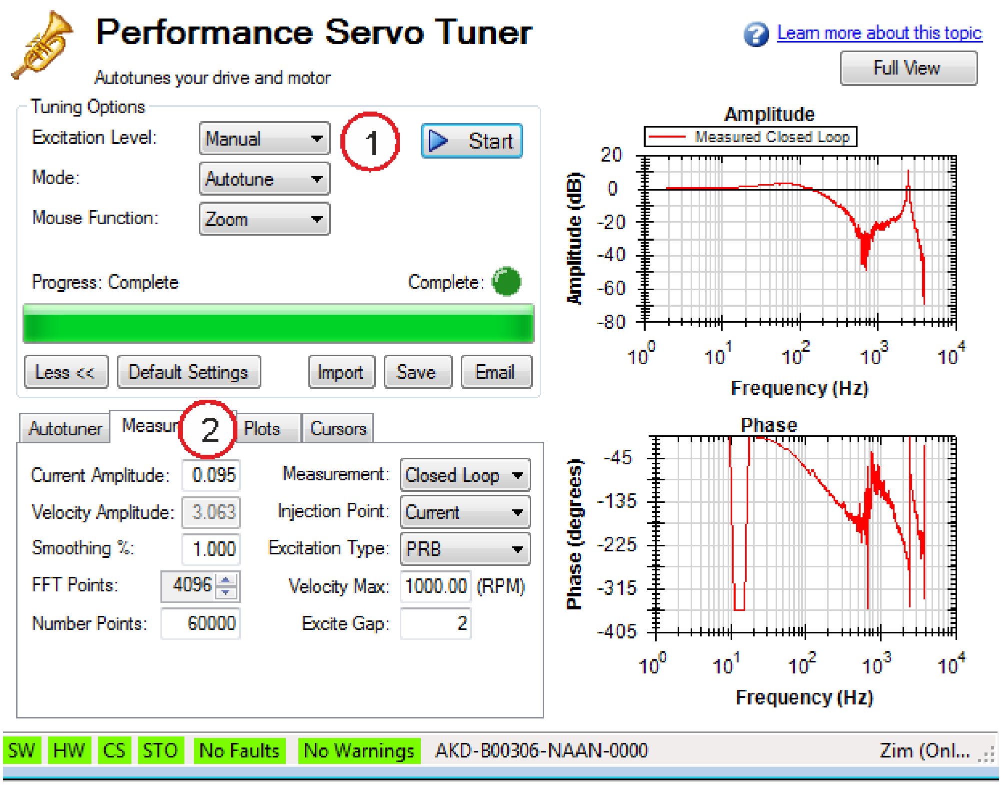
The PST also allows you to enter a manual excitation level or to take only Bode measurements (without autotuning the system).

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.

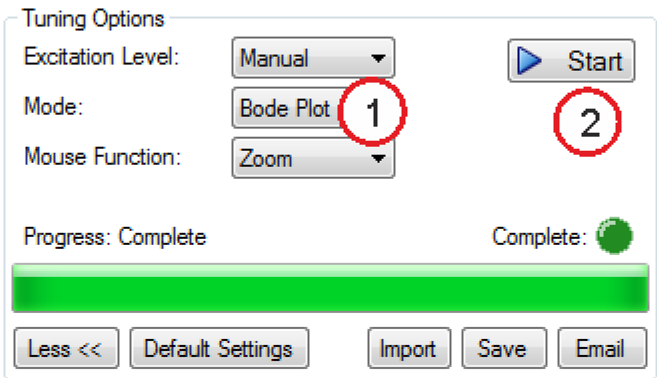
To change this excitation level, click on the **Excitation Level** drop-down box (1), and select **Manual**. Then enter a new **Current Amplitude** (2) in amps.

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.



16.3.0.4 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).



16.3.1 Using the Performance Servo Tuner: Advanced

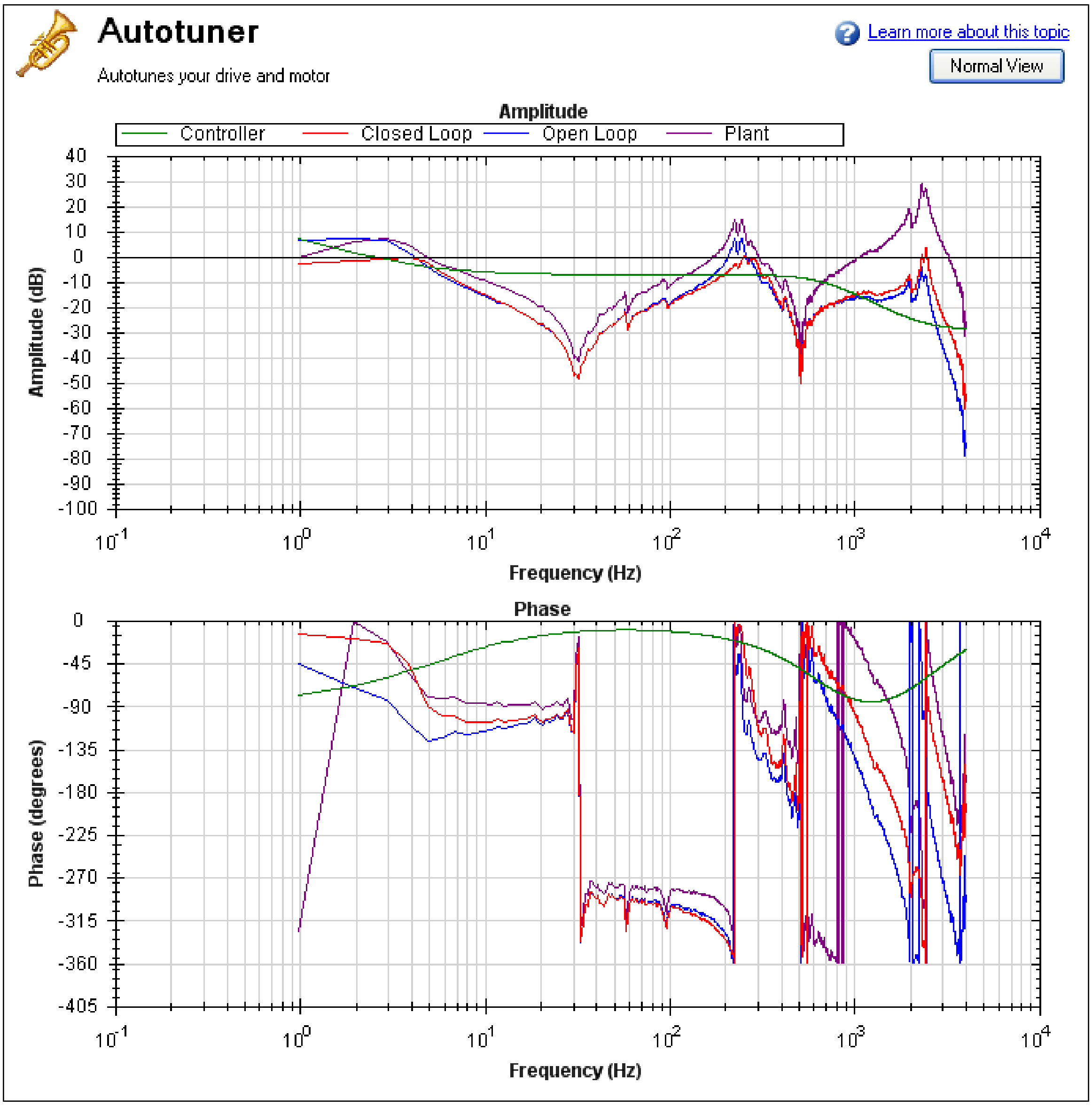
The Performance Servo Tuner (PST) can be set up to use specific modes or limits in tuning to provide tuning in ways you can control, while still taking advantage of the PST's ability to make decisions quickly and effectively for you.

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

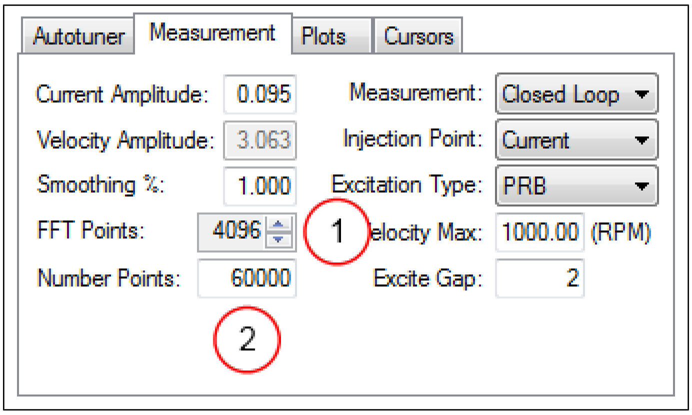
16.3.1.1 Typical Cases for Advanced PST Use

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.

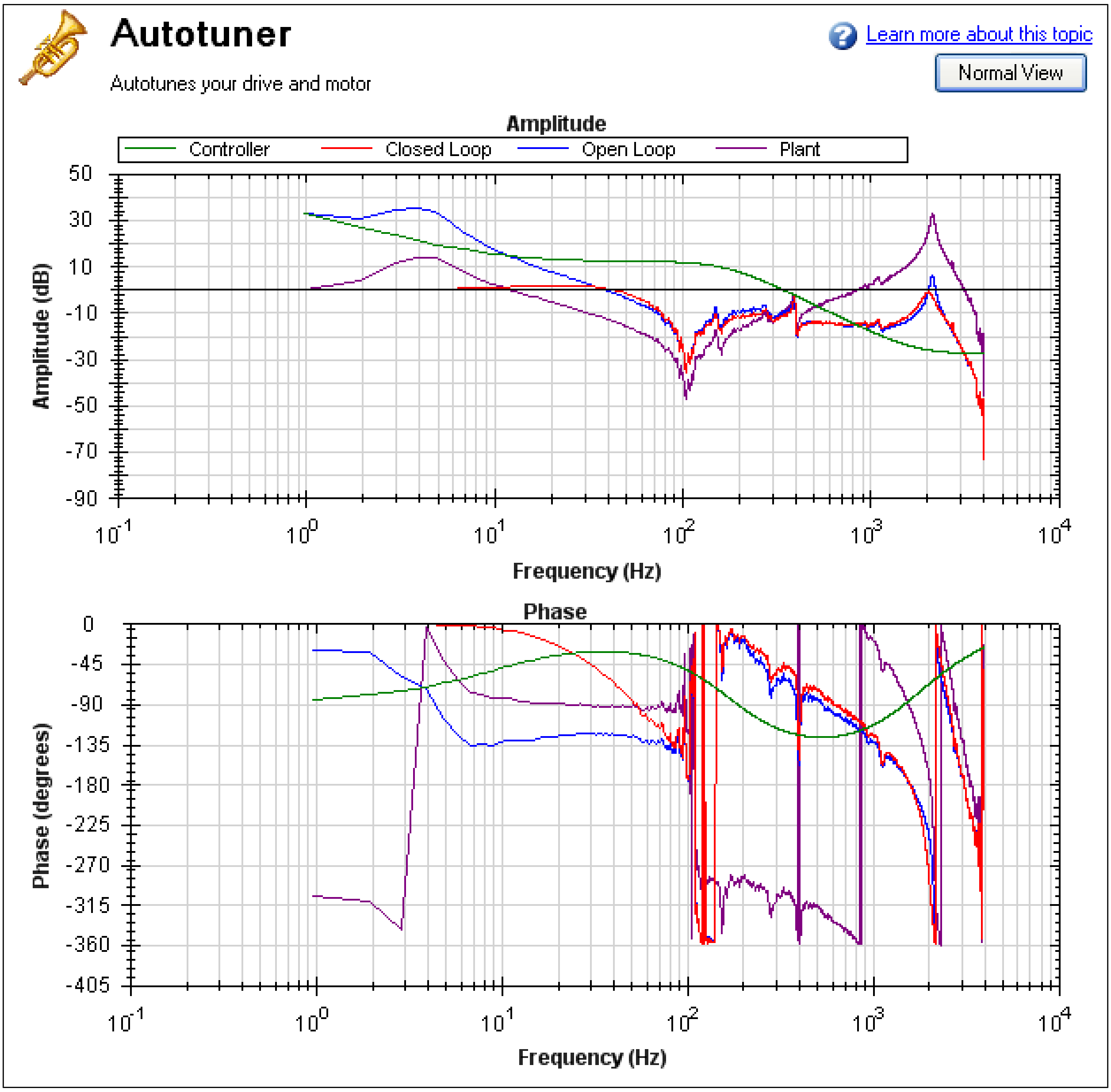


In addition, in order to accurately measure the low frequency resonances, the fast Fourier transform (FFT) resolution must be sufficiently fine to accurately measure the low-frequency resonance. A good place to start is to have an FFT resolution of 1/10 of the frequency of the lowest anti-node. In the 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.

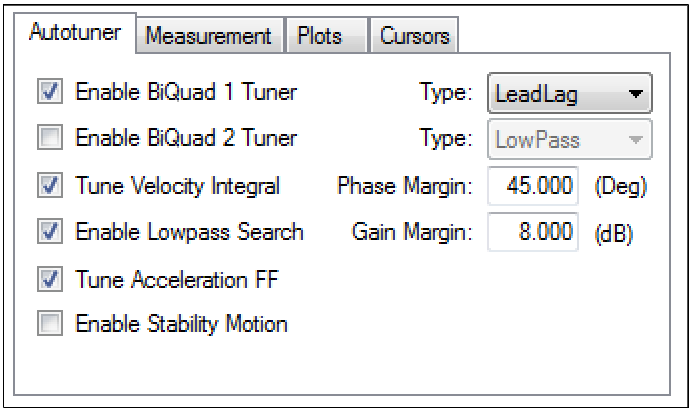


Tuning Systems with High-Frequency Resonances

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.

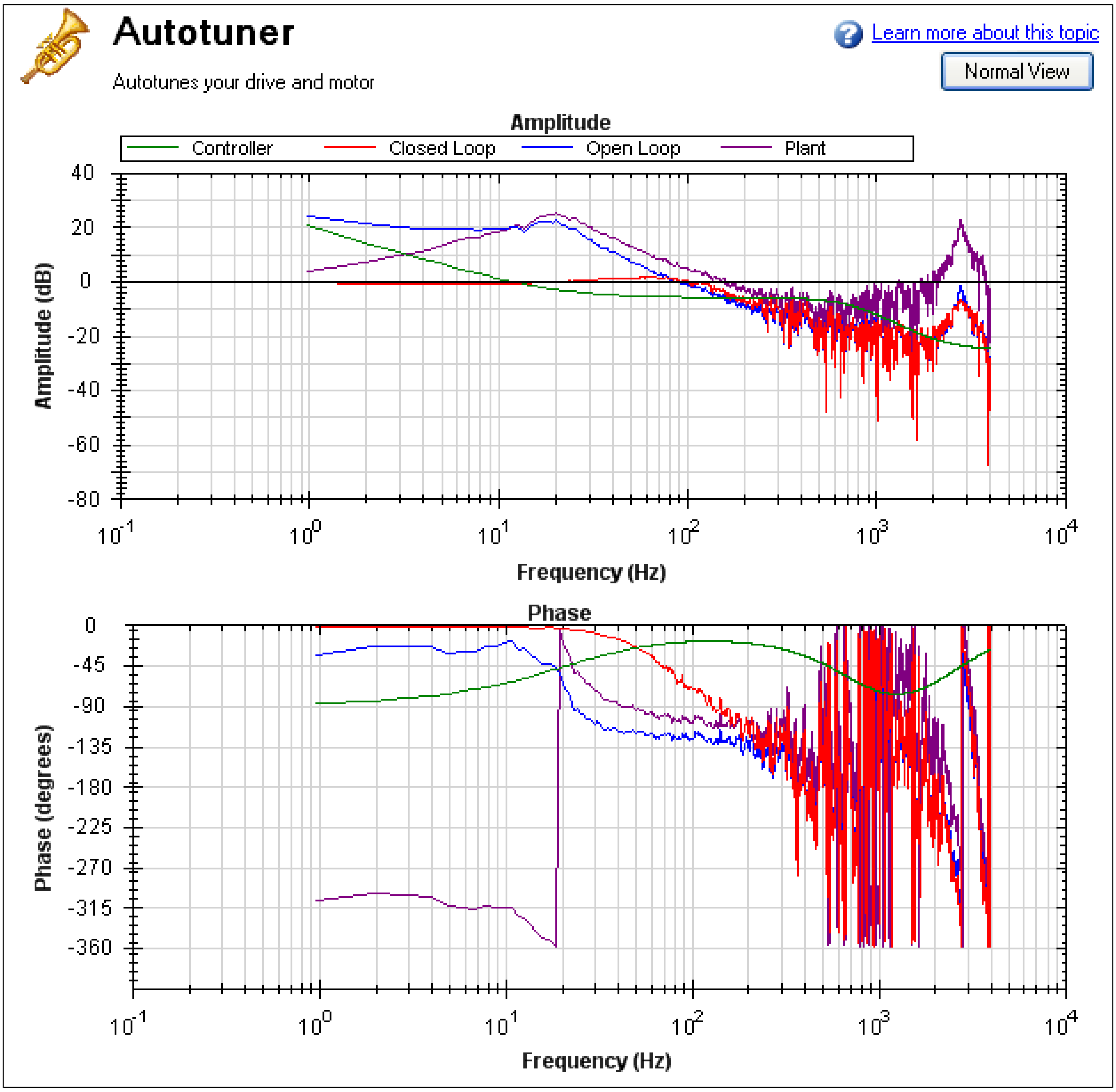


One way to resolve this problem is to use a low-pass filter in the feedback path. To use this filter, simply check the **Enable Lowpass Search** in the PST, which is the default behavior.

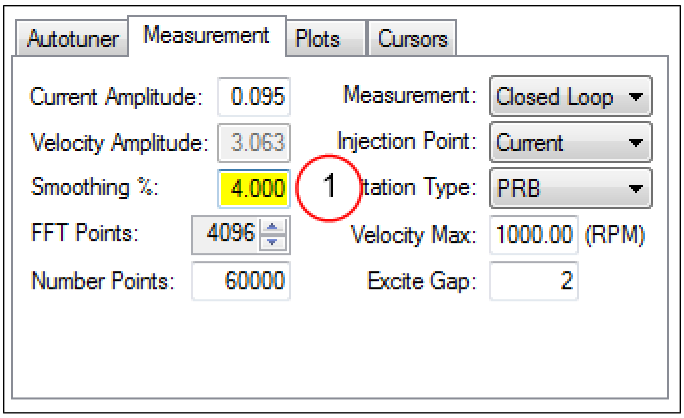


Tuning systems with noisy frequency responses

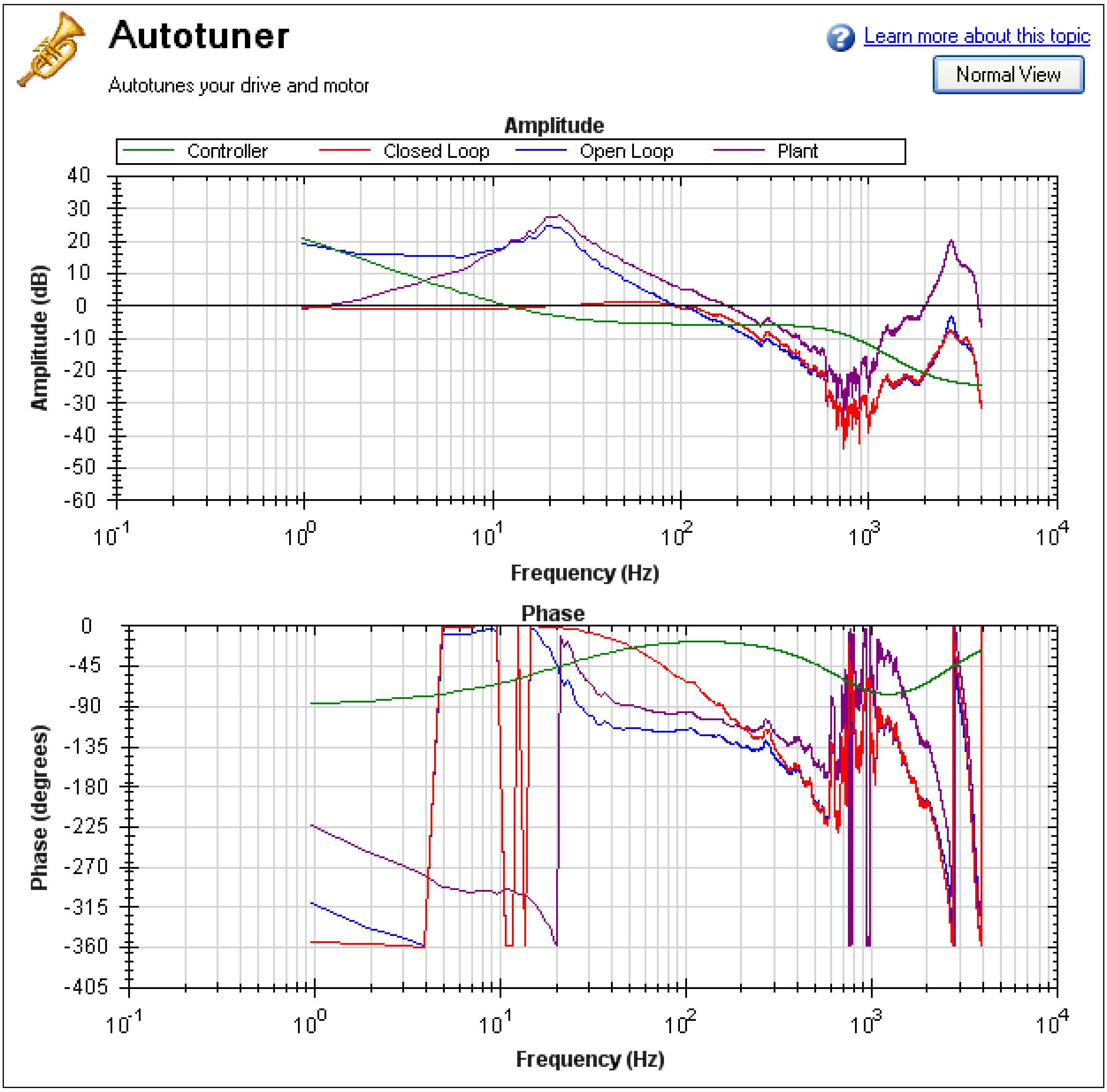
When using a motor with a low-resolution incremental encoder or resolver, the high frequency response may be noisy. Below is a Bode plot created after autotuning of an incremental encoder with 8,192 counts per revolution.



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

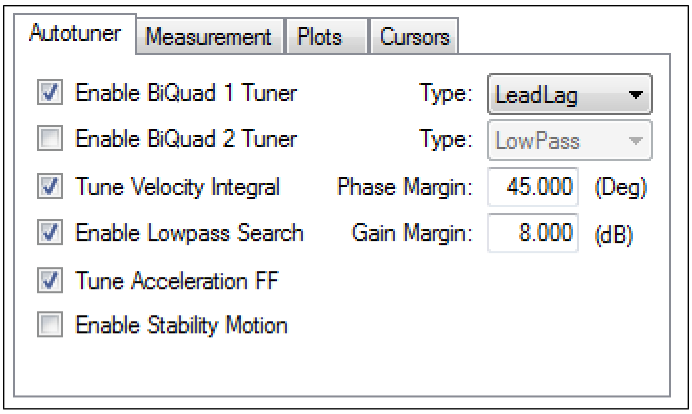


After increasing the smoothing percentage, the Bode plot traces become cleaner and easier to read:



16.3.1.2 PST Options

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



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 lowpass 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.KACCFF (pg 701))

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 (pg 251): 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

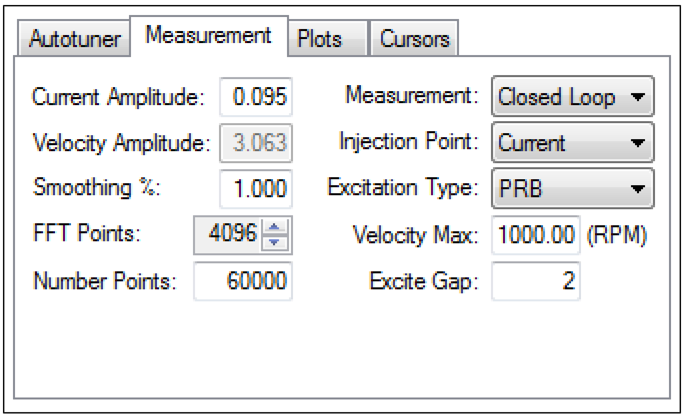
Check this box to tune VL.KI (velocity loop integral gain). If this box is unchecked, the PST will set VL.KI to zero.

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.

16.3.1.3 Measurement Options

The PST screen also provides options for measurements:



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

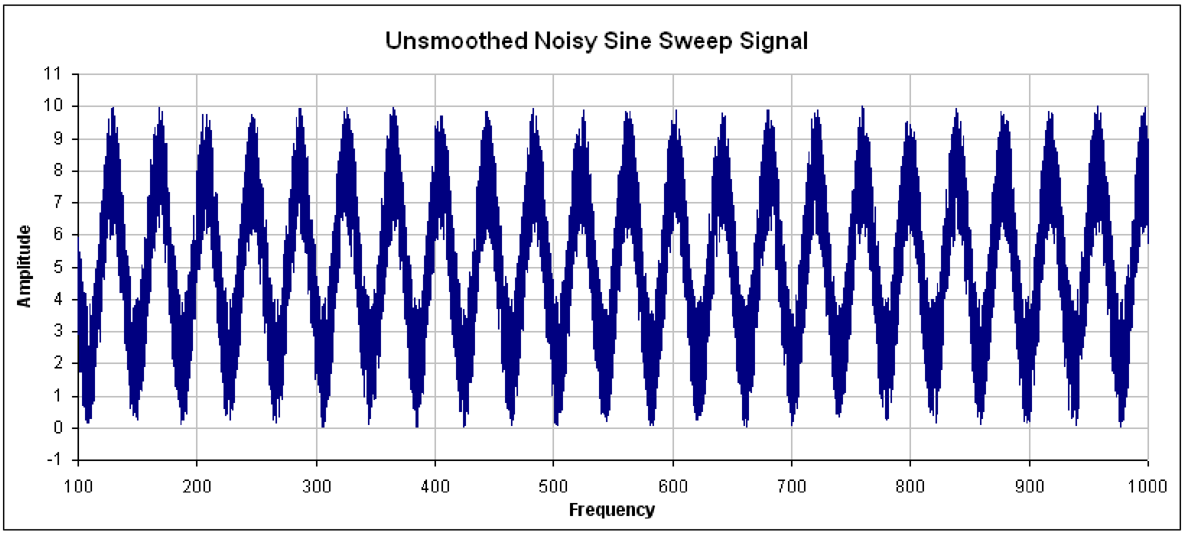
This box sets the amplitude of the velocity used to excite the system during a velocity injection mode excitation. This amplitude applies to all excitation types when the **Injection Point** is set to **Velocity**. The **Velocity Amplitude** box is disabled if the **Injection Point** is set to anything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

Smooth %

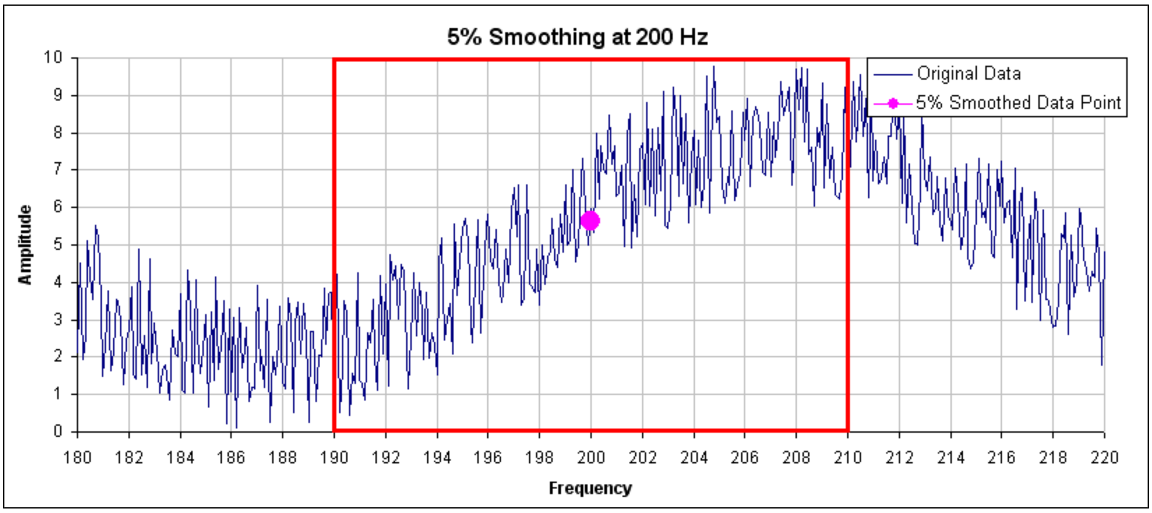
This value applies a moving average smoothing filter to the frequency response gathered during autotuning. This process reduces noise in the frequency response that can occur when making short frequency response measurements, using low resolution encoders, conducting low amplitude frequency response 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.

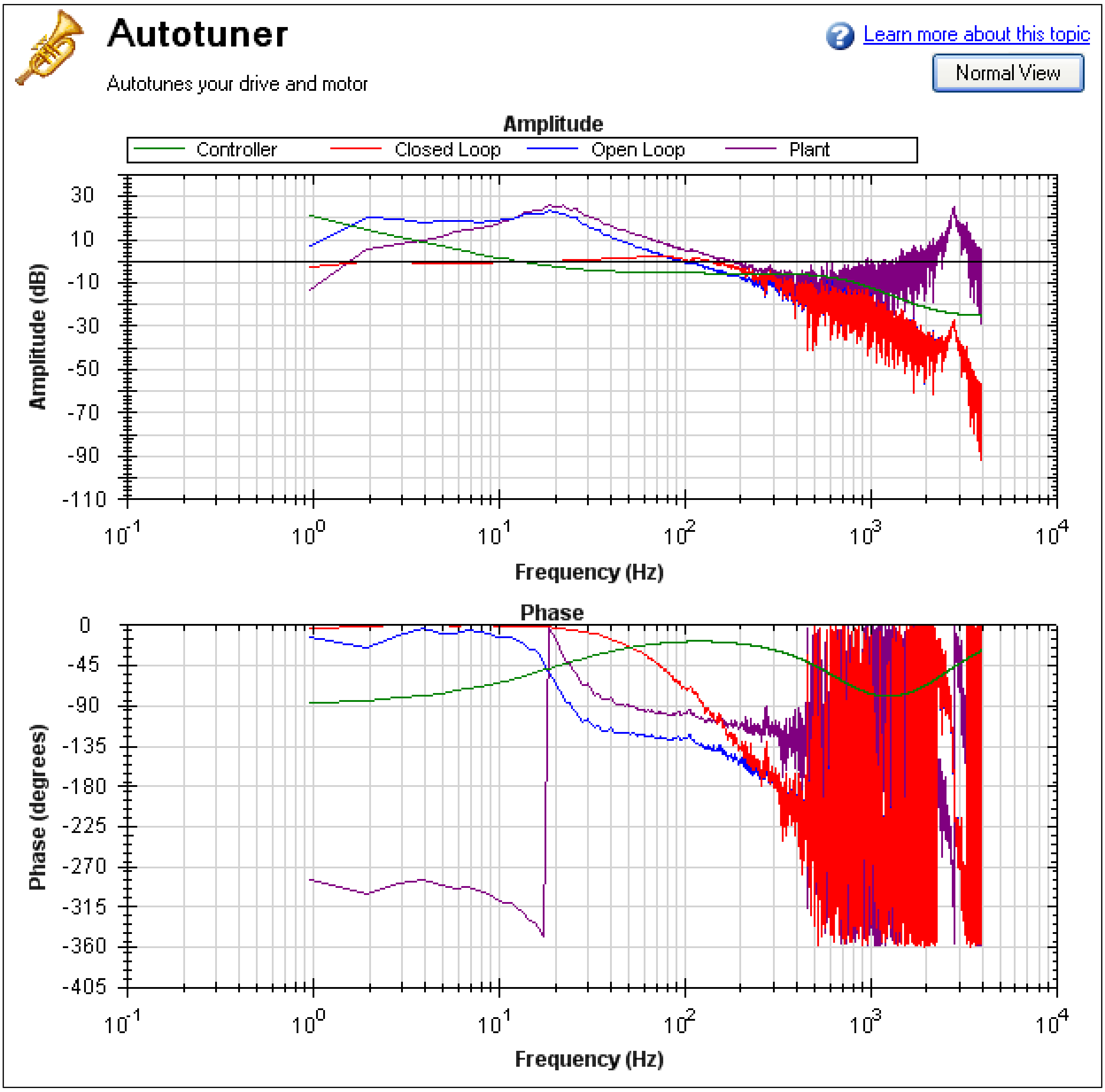


In this example, examing how the smoothing filter affects a single point shows how the smoothing filter works on a full plot. If you zoom in on 200 Hz +/- 5%, this gives a range of 190 Hz – 210Hz. The smoothing filter averages this range of values and puts the average right on 200 Hz. The figure below 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).

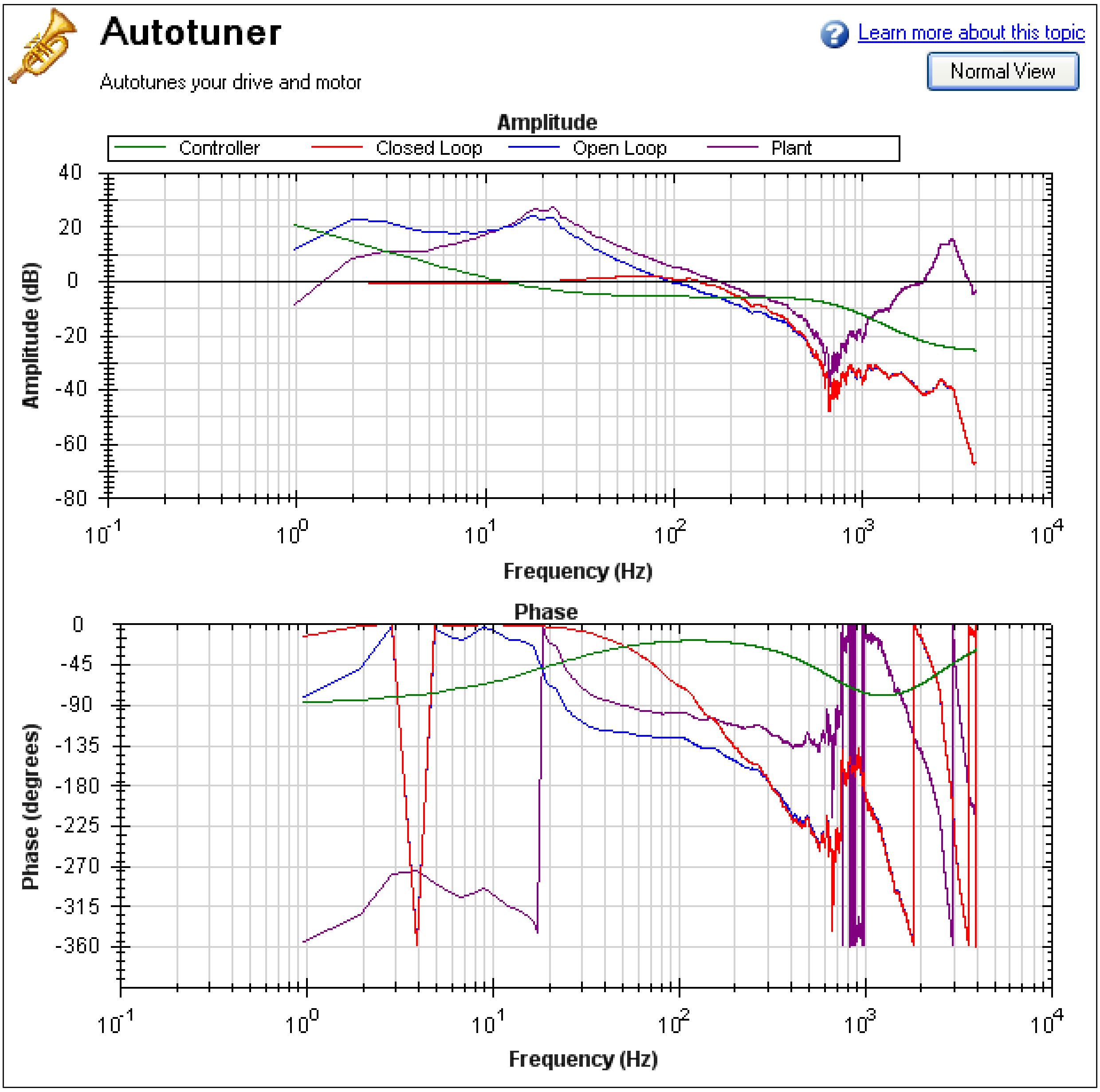


In the PST, the smoothing filter will do this analysis for every frequency point on the Bode plot. If the data is too noisy, then you can increase the smoothing percentage to smooth the noise out and see the underlying data patterns. A comparison of a system with 0.1% smoothing and 8% smoothing is shown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

0.1% smoothing


Autotuner
Autotunes your drive and motor
Learn more about this topic
Normal View
Amplitude
Controller
Closed Loop
Open Loop
Plant
Amplitude (dB)
30
10
-10
-30
-50
-70
-90
-110
10-1
100
101
102
103
104
Frequency (Hz)
Phase
Phase (degrees)
0
-45
-90
-135
-180
-225
-270
-315
-360
-405
10-1
100
101
102
103
104
Frequency (Hz)

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/(Excite Gap * Number Points) Hz

  to a higher limit equal to:

  (16,000/Excite Gap) 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^BODE.PRBDEPTH (pg 424) * Excite Gap)) or 16,000/(Excite Gap * Number Points) Hz

  to a higher limit equal to:

  (16,000/Excite Gap) Hz

  BODE.PRBDEPTH (pg 424) 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/(Excite Gap * 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.

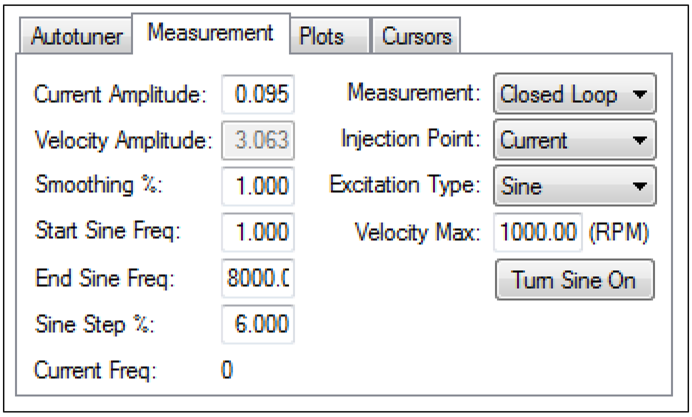
Number Points

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:

Number Points * Excite Gap/16,000 seconds

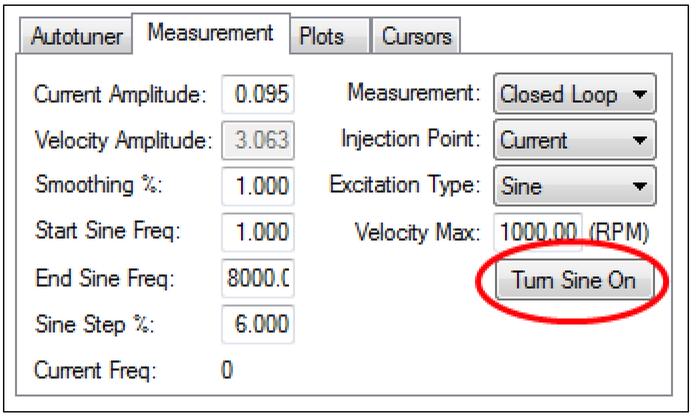
Velocity Max

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 (pg 955)) is restored.



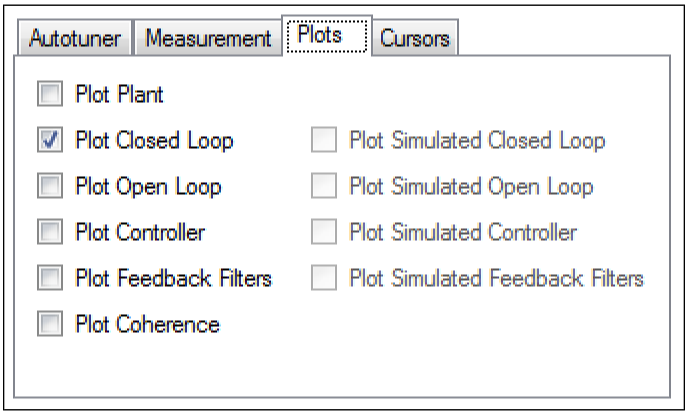
If **Excitation Type** box is set to **Sine**, different configuration options become available.

- **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.



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.

16.3.1.4 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:

$$Coherence = \frac{(Pxy \times Pxy^*)^2}{(Pxx \times Pxx^*) + (Pyy \times Pyy^*)}$$

where:

Pxx = Power Spectral Density of Input signal

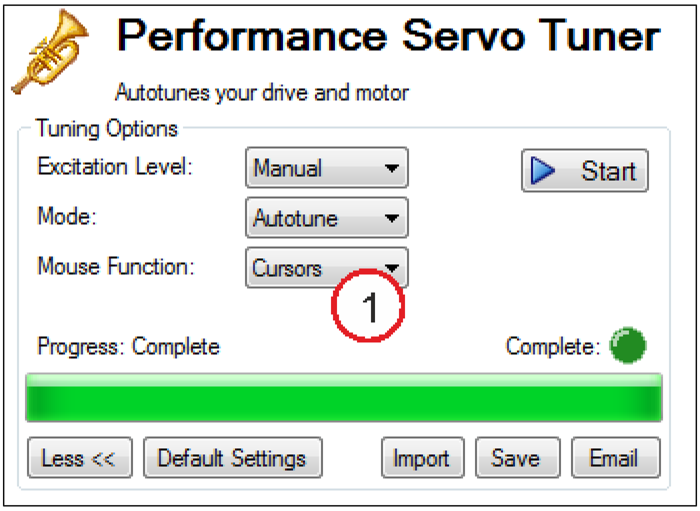
Pyy = Power Spectral Density of Output signal

Pxy = Cross Spectral Density of Input and Output

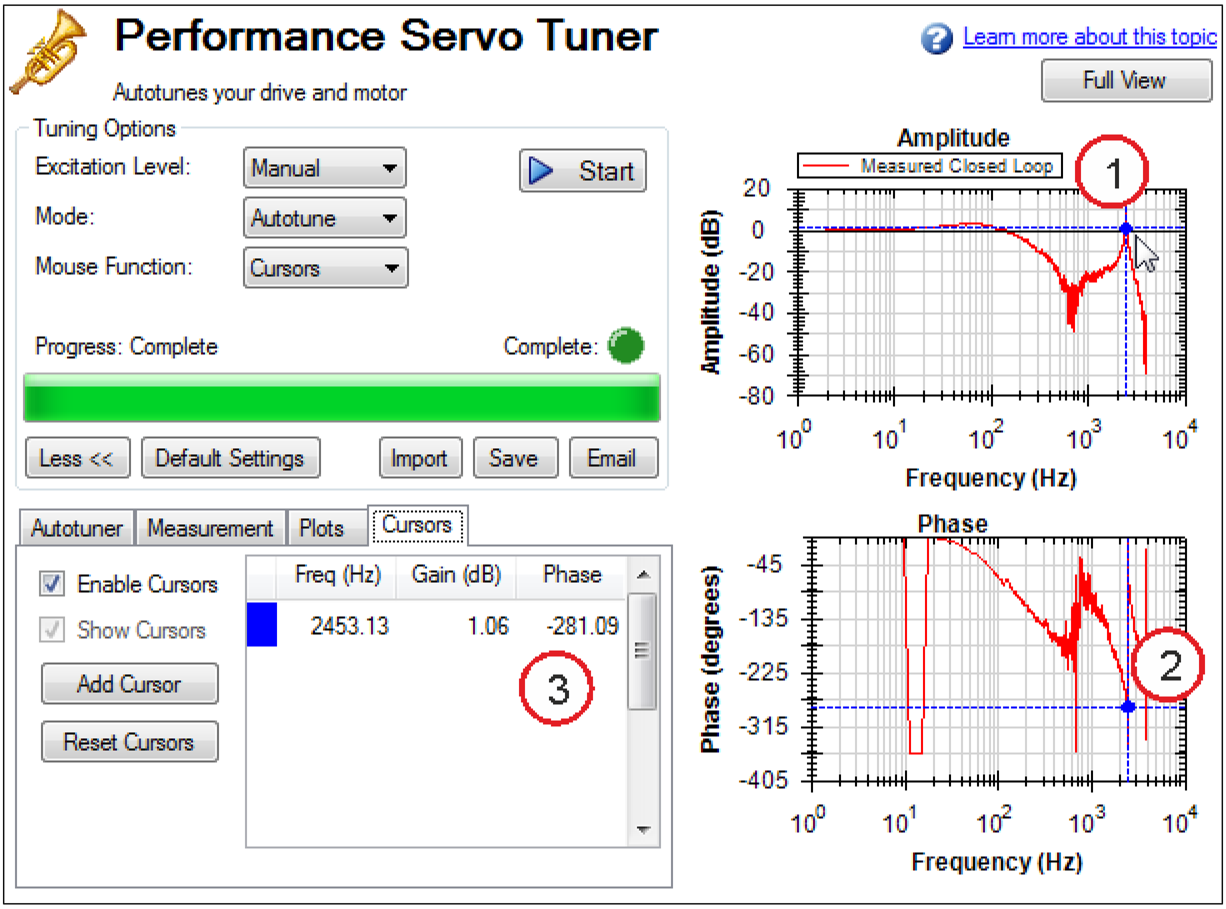
* designates complex conjugate

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 dropdown(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.

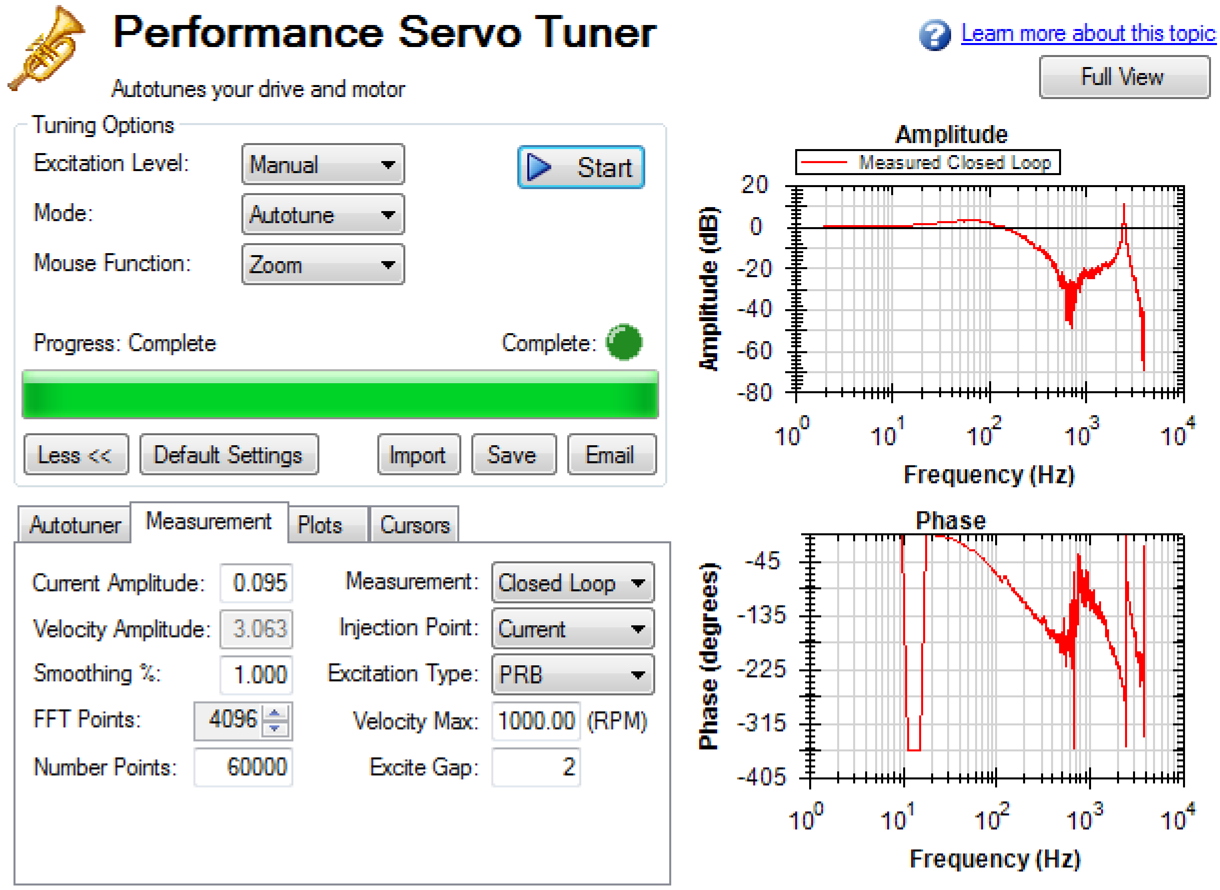
The dotted crosshair lines are only drawn for the active cursor selected; to remove all cursors from the screen, but retain their position, uncheck **Show Cursors** . To reset all cursors, click **Reset Cursors**.

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.

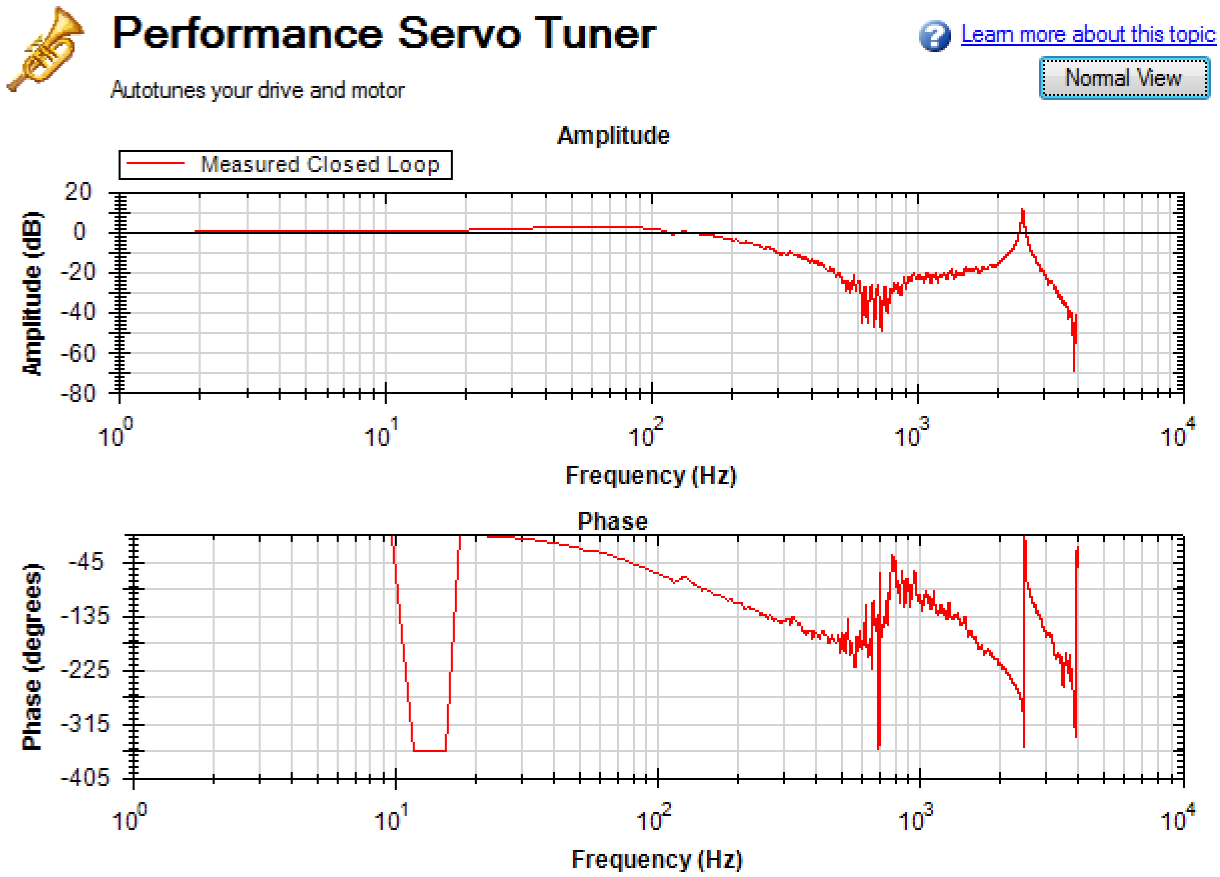
16.3.1.5 Resizing Bode Plots

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



Simple measurement full view



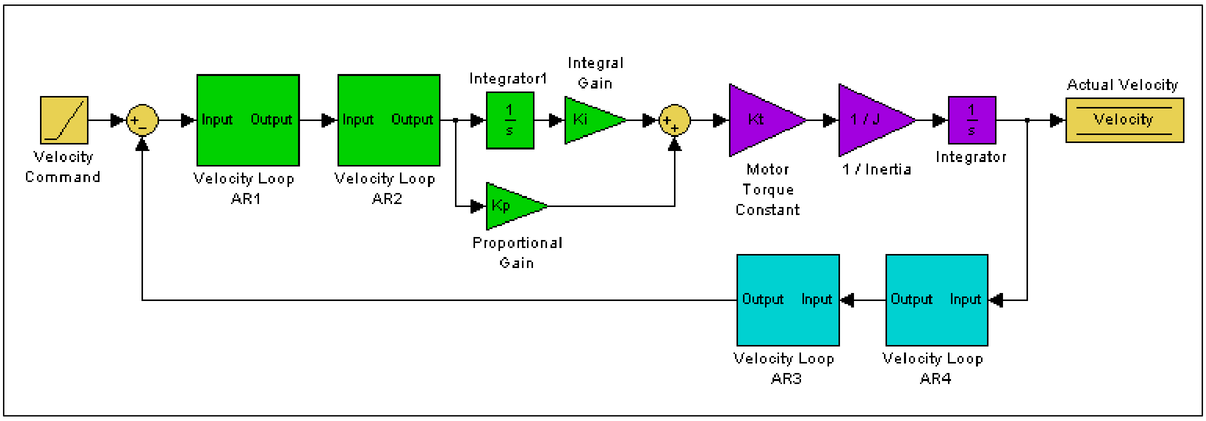
Reading and Understanding the Bode Plot

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 (pg 212) documentation.

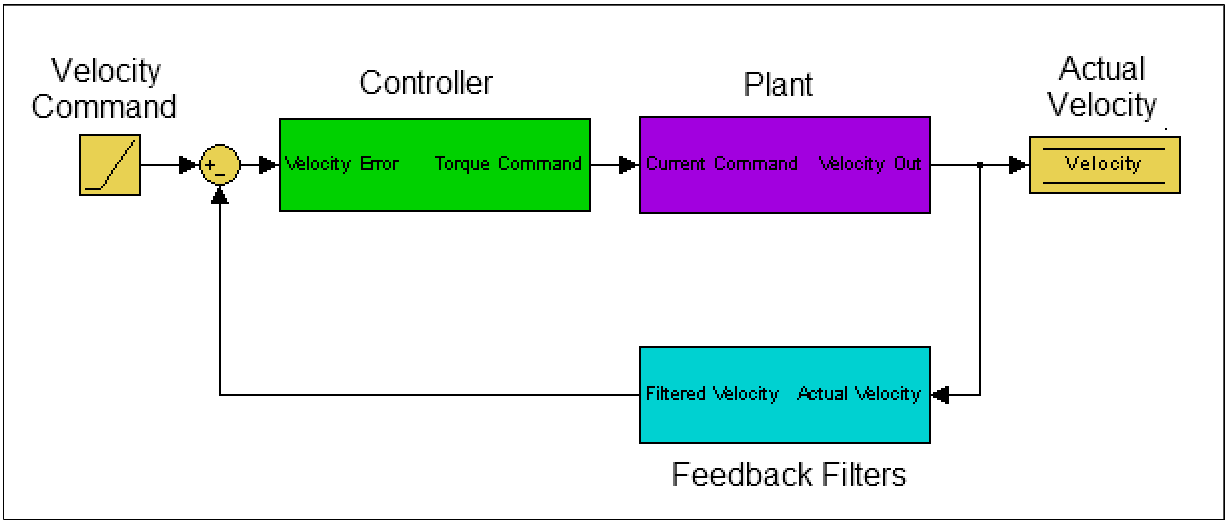
Four Bode plot traces are displayed by default:

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 (pg 212)



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:

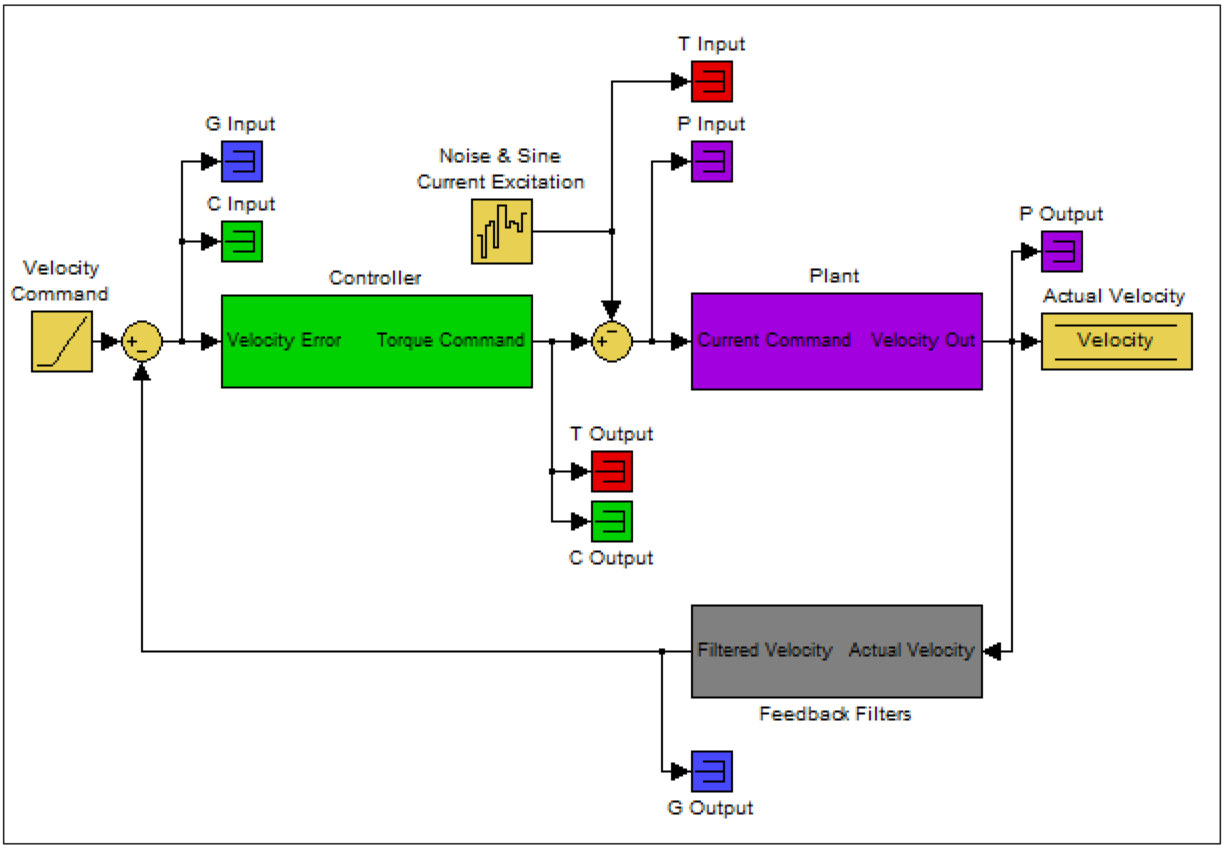
Open Loop = Controller x Plant x Feedback Filters

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

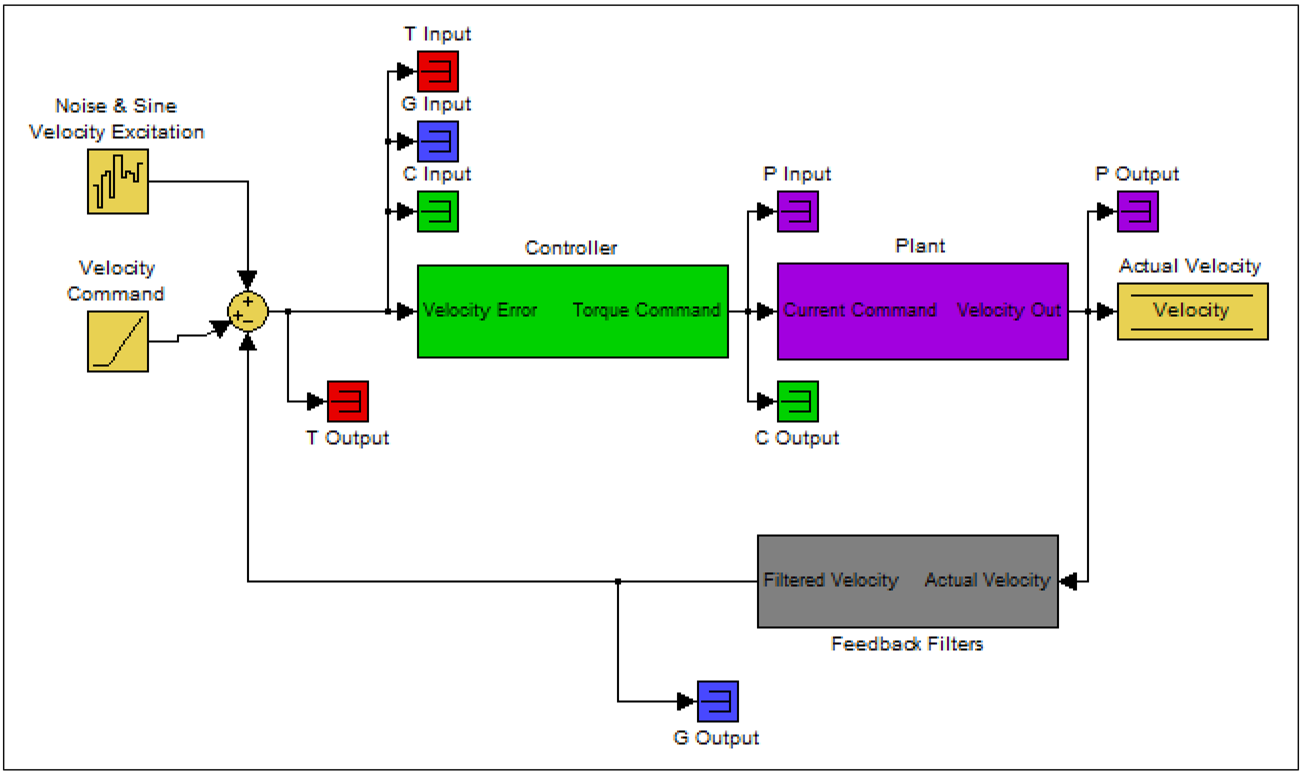
$$ClosedLoop = \frac{Controller \times Plant}{1 + Controller \times Plant \times FeedbackFilters}$$

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 (pg 212) 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.

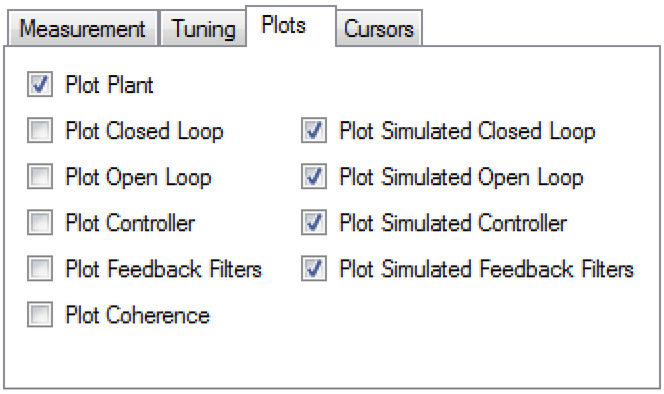
Several differences exist between PST and Bode Plot Interfaces:

- 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:

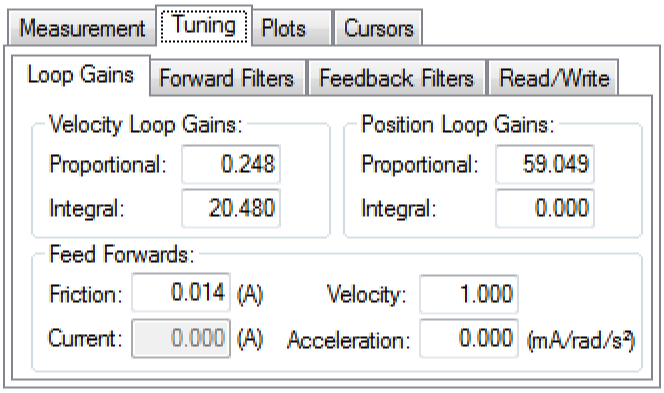


These selected boxes are the most common configuration for tuning; however, simulation will occur regardless of the checkboxes selected .

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.

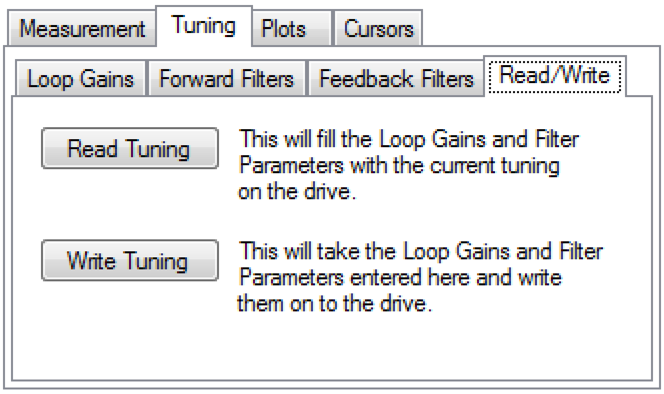
Using the Performance Servo Tuner Manual Tuning Interface

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.

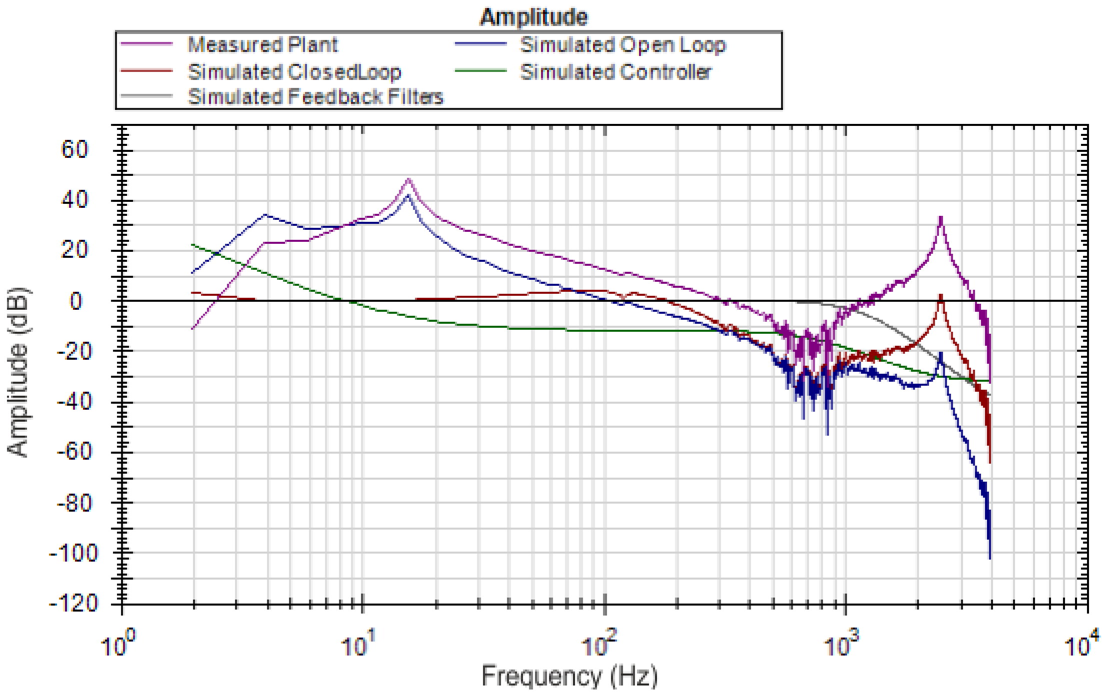


To restore the tuning on the drive to the PST interface, click the **Read Tuning** button.

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.

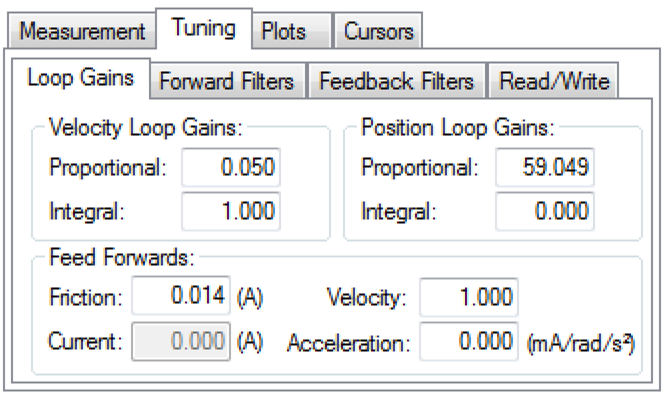
Simulating Modified Loop Gains with the Performance Servo Tuner

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

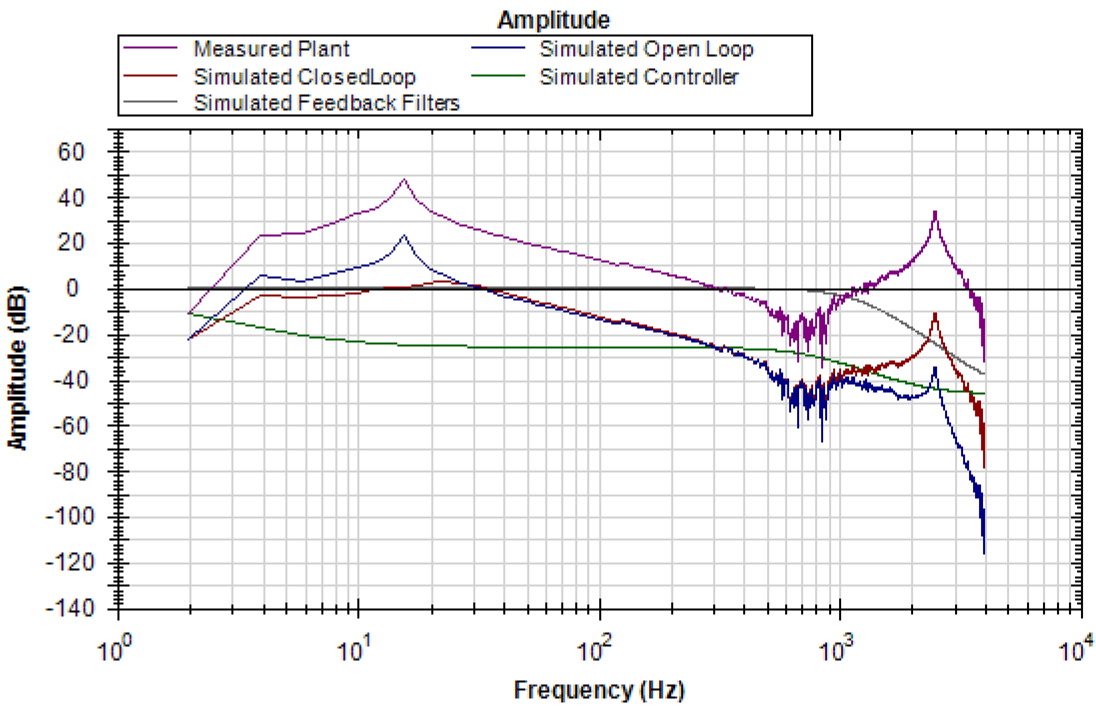


The Velocity Loop Proportional gain here is 0.248. If an application did not need to be tuned as stiff as this, then you could use the PST simulator to detune the motor to the desired bandwidth. A followup Bode Measurement can verify that the simulated response is correct.

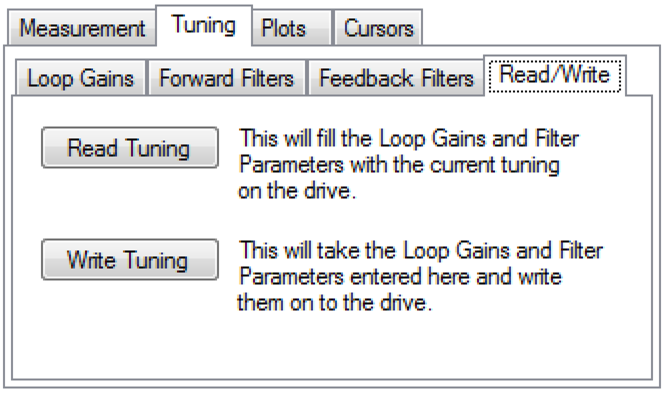
Use the boxes in the **Loop Gains** tab to change tuning gains until the desired frequency response is achieved.



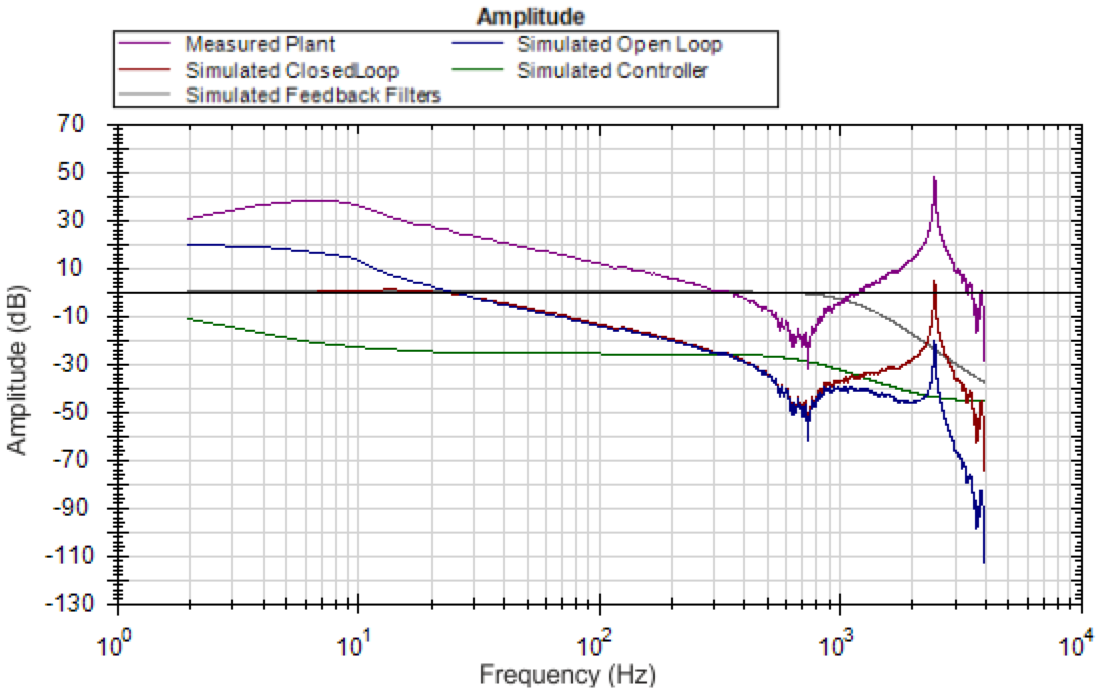
The de-tuning of velocity loop proportional and integral gains simulated that the bandwidth of the servo has been detuned from ~100 Hz to ~30 Hz.



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.



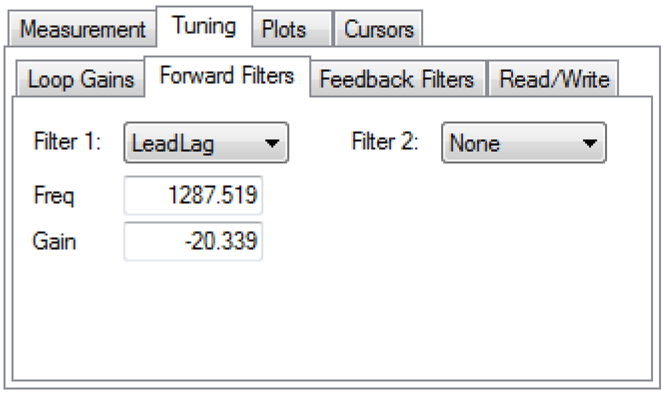
The new measured Bode Plot indicates we achieved slightly lower than 30 Hz bandwidth. The servo is stable, and tuning can be refined until desired performance is reached.

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.

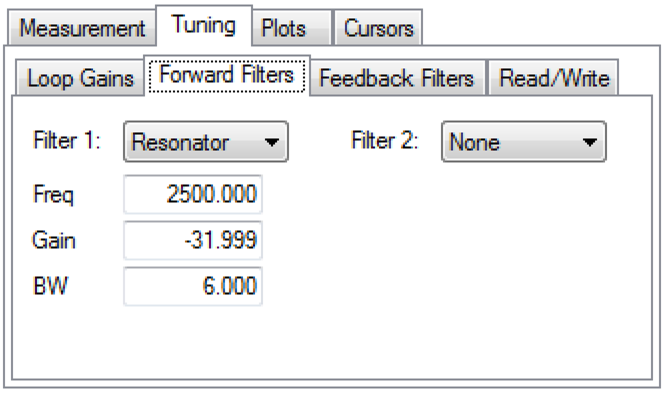
Click on the **Forward Filters** tab:



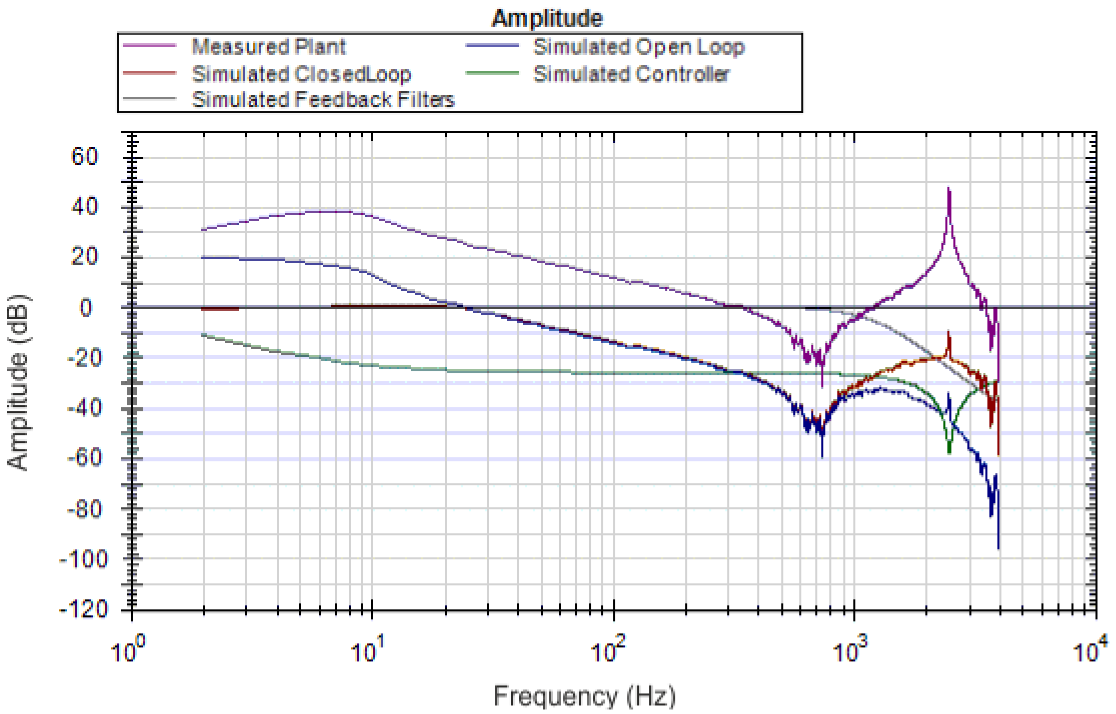
The results of the autotune are still on the drive, and provide adequate tuning. A lead lag filter is the default tuning filter, and is a good general case filter for most servo loops.

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.

16.4 Tuning Guide

16.4.1 Overview

This section covers tuning the velocity and position loops in the AKD. 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 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 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, 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.

16.4.2 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:

    a. +/– x position error counts during the entire motion.
    b. Settling within +/- x position error counts, within y milliseconds.
    c. 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.

16.4.3 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.

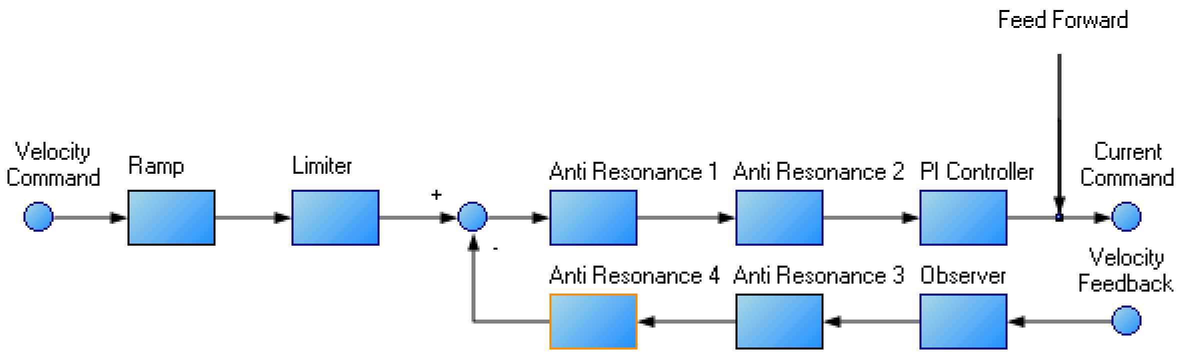
16.4.4 Closed Loop Tuning Methods

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.

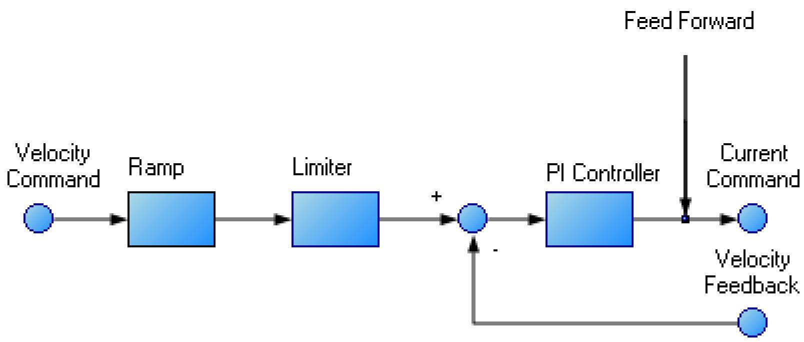
When in velocity operation mode, only the velocity loop is tuned. When in position operation mode, both the velocity and position loops must be tuned.

16.4.4.1 Tuning the Velocity Loop

The velocity loop on the AKD 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.

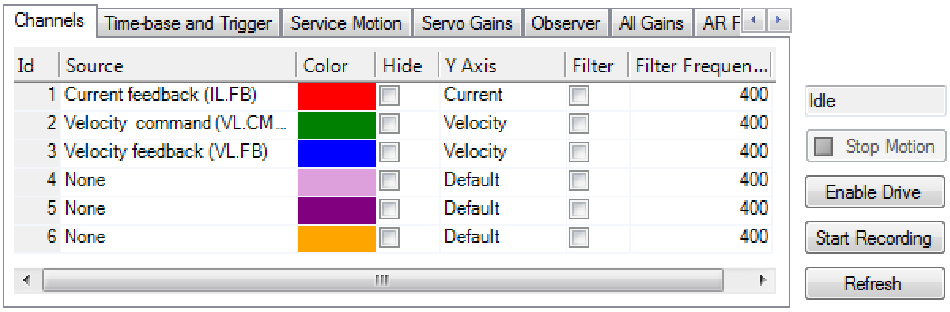


To perform basic tuning of the velocity loop, you can use just the PI block and set ARF1 and ARF2 to unity (no effect) and set the observer to 0 (no effect). Using just the PI block simplifies the process of tuning the velocity loop. To start tuning you can adjust the PI Controller block first. A simplified 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 ½ 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 velocit loop.

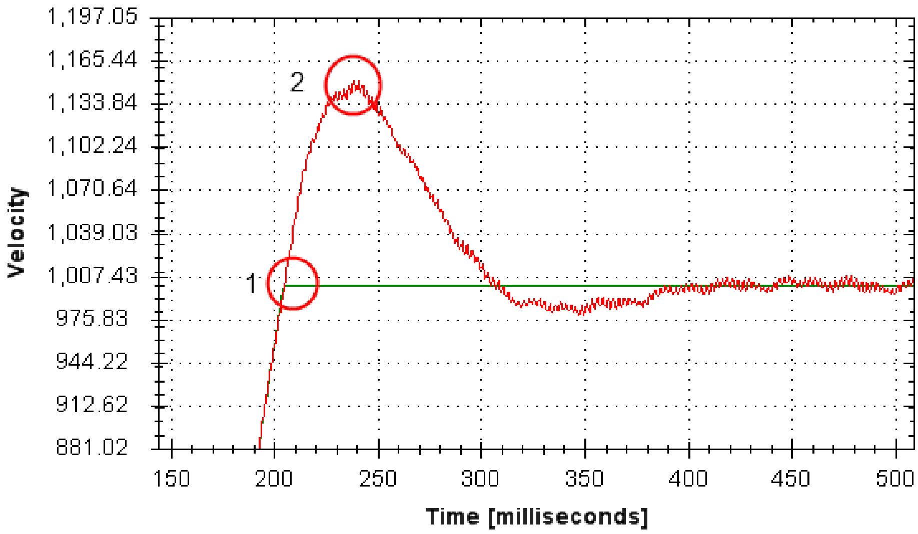


7. Adjust VL.KP. Keep increasing VL.KP by a factor of 2 until you either:

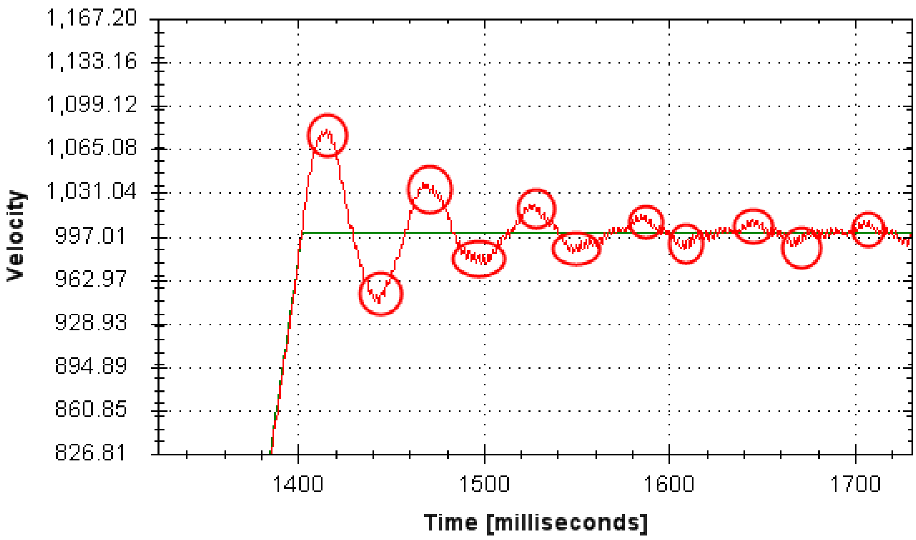
- Hear an objectionable noise from the system (buzzing, humming, etc) or
- See velocity overshoot. No velocity overshoot should be present when using only VL.KP.
- When you reach one of the limits above, decrease VL.KP to the value where there were no objectionable noises or overshoot.

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.



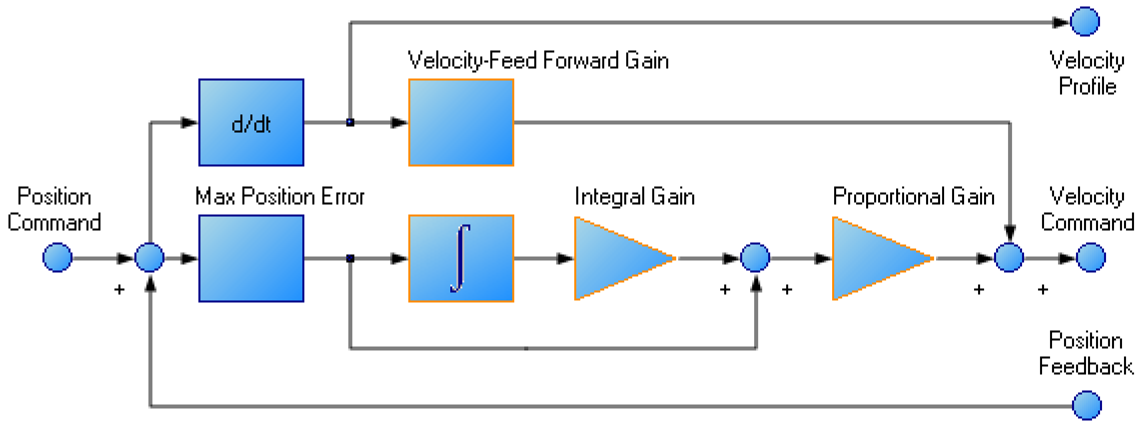
- When you reach one of the limits above, decrease VL.KI to the value where there were no objectionable noises or overshoot.

9. Stop the service motion

16.4.4.2 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:

1. Set VL.KVFF to 1
2. 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.

16.4.5 Torque Feedforward Tuning Methods

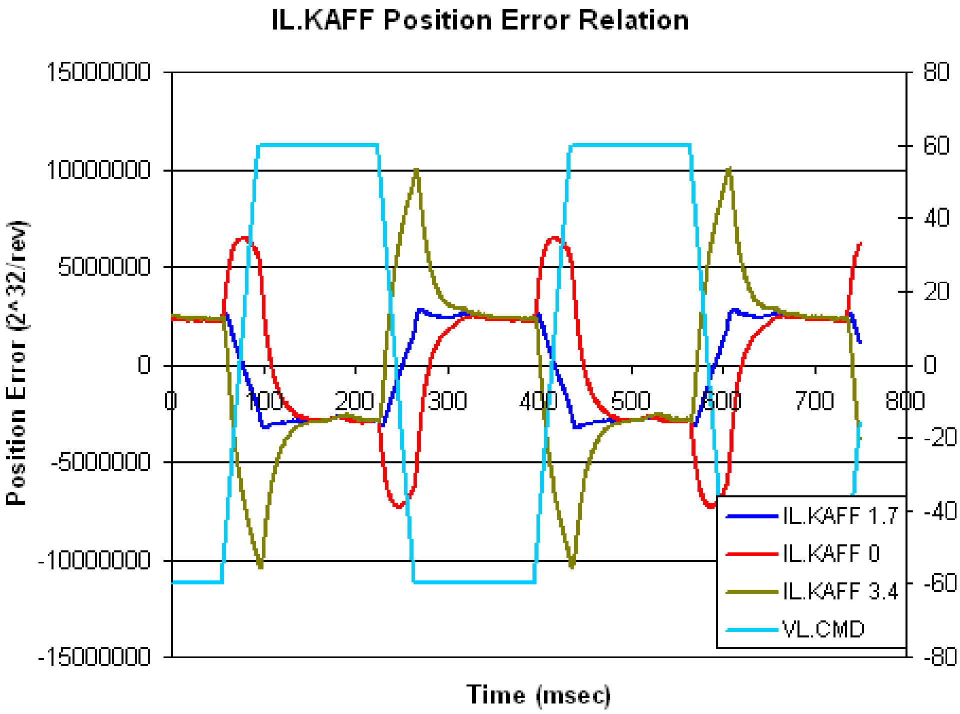
The torque based feedforward terms on the AKD effectively model the physics of your motor and allow the drive to command the appropriate current, even before the encoder has time to send data back to the drive. Torque based feedforward terms allow you to lower following error with virtually no stability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

16.4.5.1 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 decel-

eration. The picture below has an ideal value of IL.KAFF of 1.7.



16.4.6 Using Anti-Resonance Filters

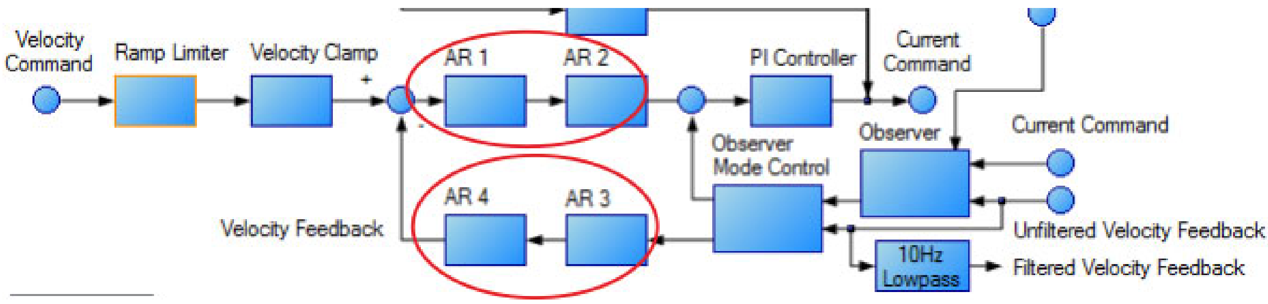
The AKD has four anti-resonance filters. Two filters are in the forward path and two are in the feedback path.

Similarities

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

Differences

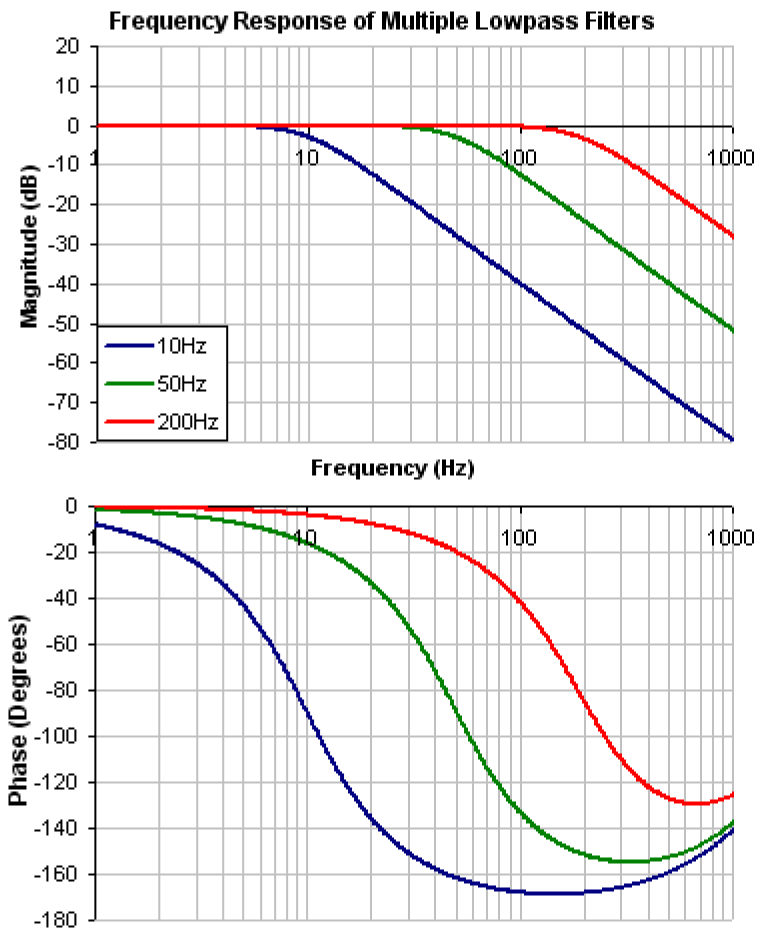
- Forward path filters result in higher phase lag in closed loop system response.
- Forward path filters limit spectrum from reaching the motor / feedback path filters only filter the feedback after it has been to the motor.



16.4.6.1 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-spass 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

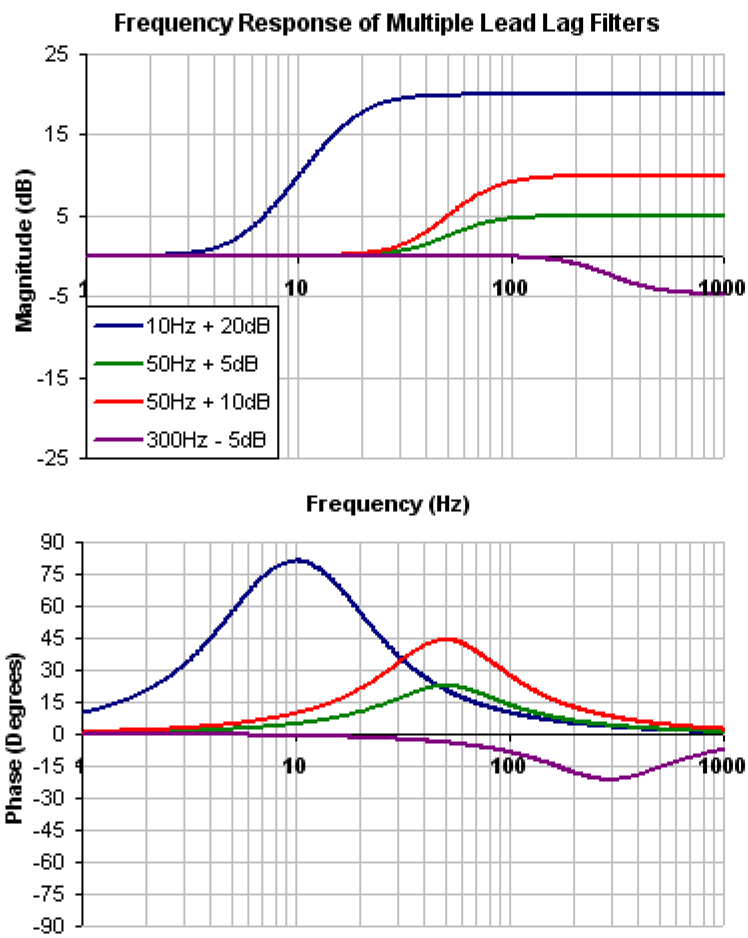
VL.ARZQ1 0.707

VL.ARPF1 5000

VL.ARPQ1 0.707

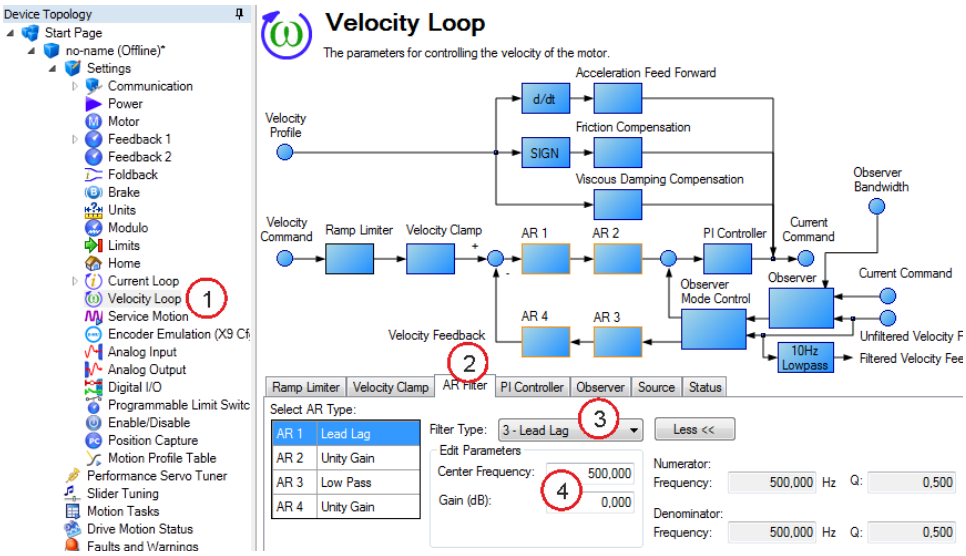
Lead Lag

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.



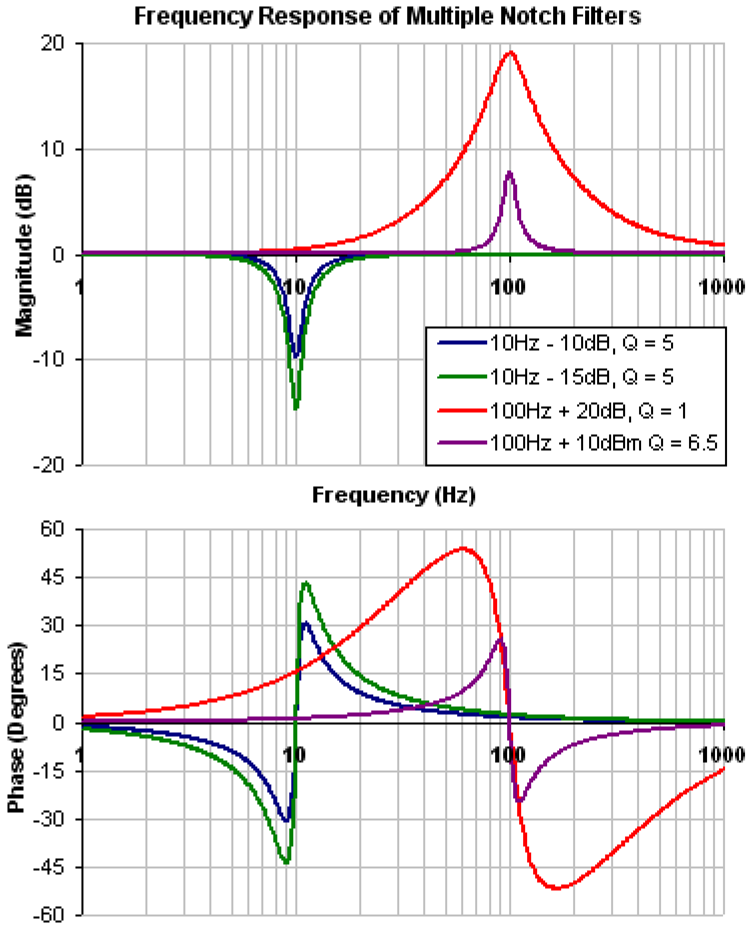
To specify a Lead Lag filter, you must specify the Center Frequency and high frequency Gain (dB). To do this, see the following example by clicking on the Velocity Loop:

Click on Velocity Loop (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).



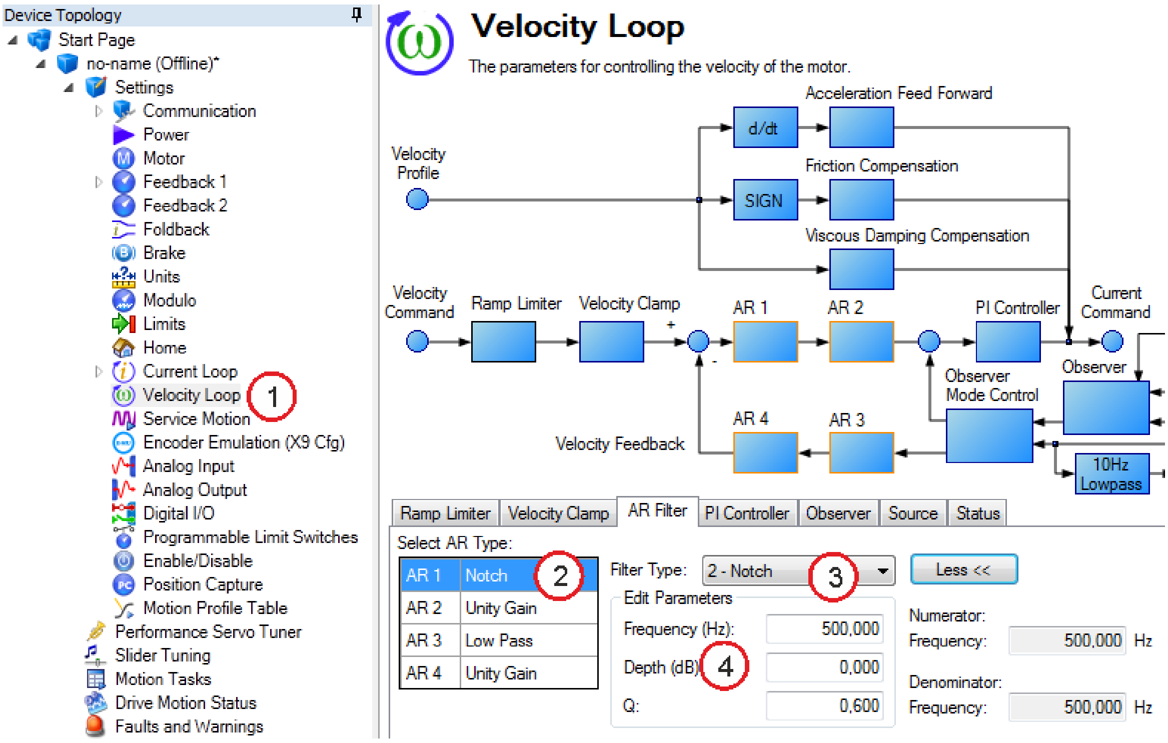
Notch

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).



Biquad

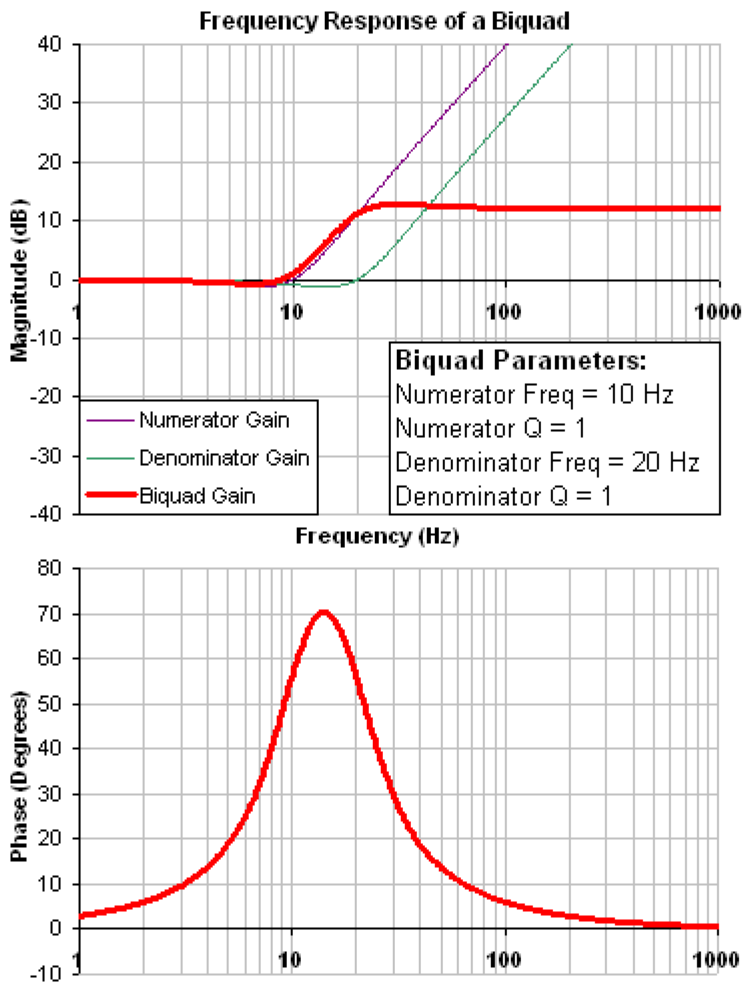
A biquad is a flexible filter that can be thought up as being made up of two simpler filters; a zero (numerator) and a pole (denominator). In fact, the pre-defined filters mentioned above are really just special cases of the biquad.

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.

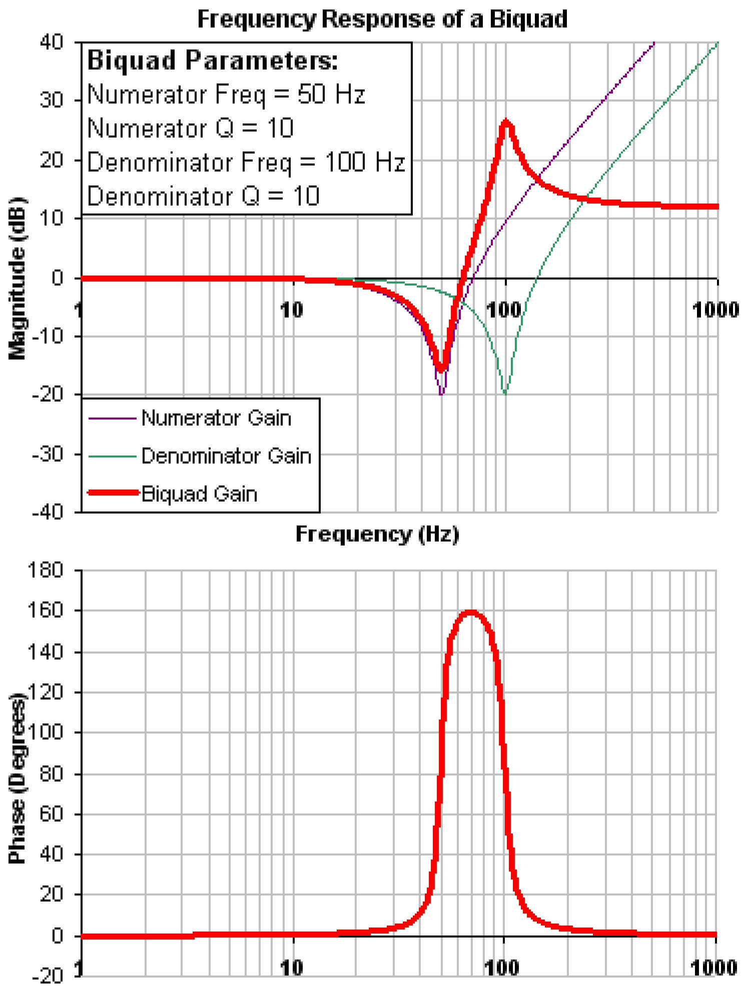
Analyzing the numerator and denominator, the frequency response calculation is simple:

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.

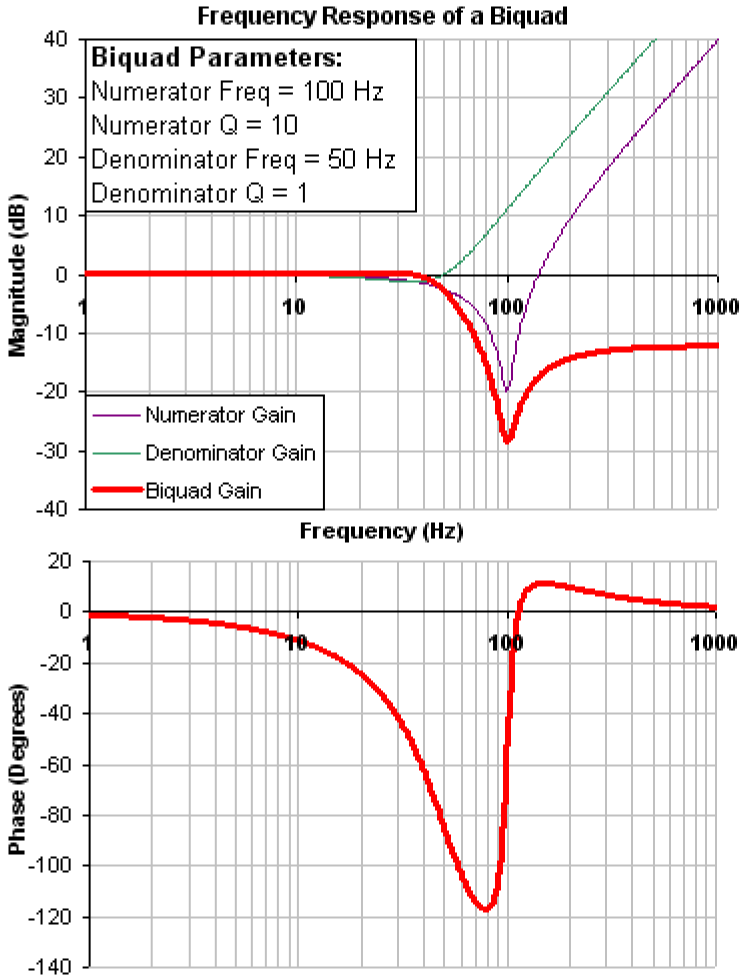


The biquad filter is very flexible, which allows custom filters to be designed. Below is an example of a resonance filter using a biquad. Notice how the high Q values affect the numerator and denominator. This gives a biquad frequency response similar to a mechanical resonance.



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

16.4.6.2 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:

$$t = 62.5\mu\sec$$

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

$$N_{2Unscaled} = 1$$

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

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

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

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

$$D_2 = 1$$

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

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

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

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

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

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

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

$$\angle_Z = \omega\, t = 2\pi * freq * t$$

$$Num_{\mathrm{Re}} = N_2 Cos(2\angle_Z) + N_1 Cos(\angle_Z) + N_0$$

$$Num_{\mathrm{Im}} = N_2 Sin(2\angle_Z) + N_1 Sin(\angle_Z)$$

$$Den_{\mathrm{Re}} = D_2 Cos(2\angle_Z) + D_1 Cos(\angle_Z) + D_0$$

$$Den_{\mathrm{Im}} = D_2 Sin(2\angle_Z) + D_1 Sin(\angle_Z)$$

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

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

16.4.6.3 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.

17 Scope

17.1 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 and drive and scope control buttons (**Enable Drive**, **Start Recording**, and **Refresh**).

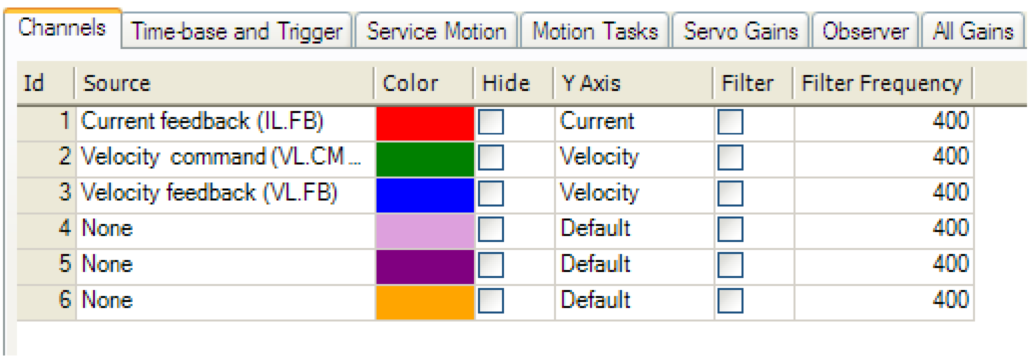
17.2 Using the Scope

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.
Service Motion	Generate basic motion.
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.

17.2.1 Scope Channels Tab

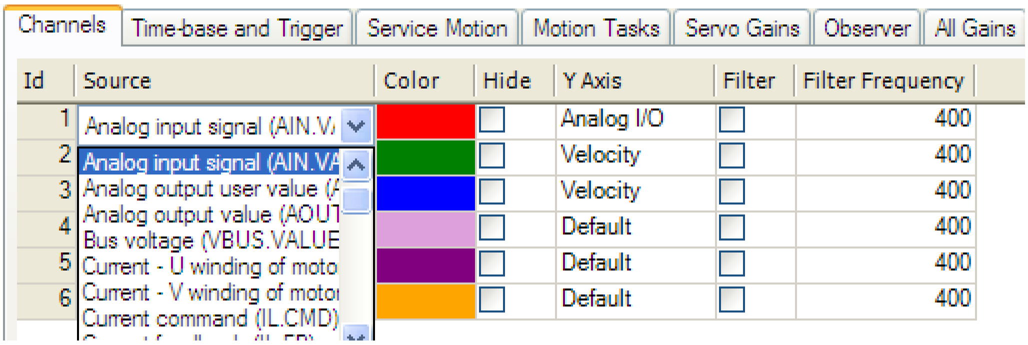
The **Channels** tab allows you to select and record up to six channels simultaneously. Select the data to record for each channel from the lists in the **Source**, **Color**, **Y-axis**, and **Filter** and **Filter Frequency** columns. Once a recording is shown on the scope screen, you can click **Hide**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



17.2.1.1 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.



17.2.1.2 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.

17.2.1.3 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.

17.2.1.4 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.

17.2.1.5 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.

17.2.2 Measure Tab

The measure tab displays basic data reads from the plots.

17.2.2.1 AC RMS

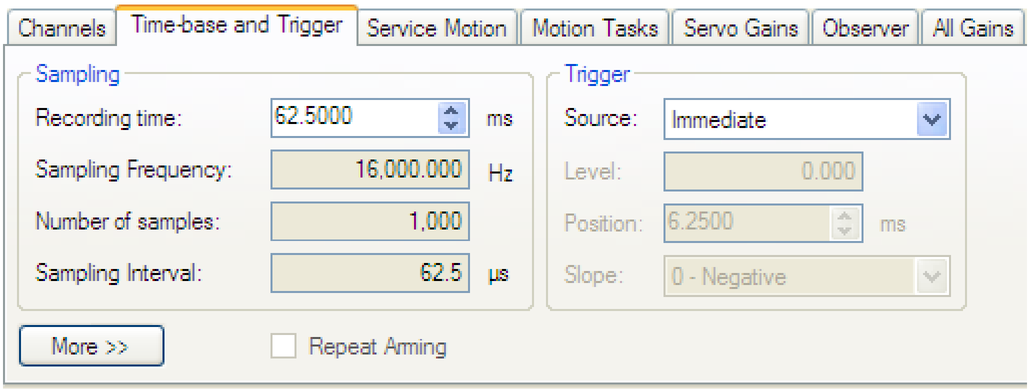
In the measure tab, the column labeled RMS displays AC RMS. True or full RMS is the full heating value of a signal and includes any DC terms in the value. AC RMS removes any DC value and gives only the RMS value as a measure of a signal's standard deviation.

```
True RMS = Sqrt{Sum(x[n]^2)/N} where N is number of points
```

```
AC RMS = Sqrt{(True RMS)^2 - (dc or average value)^2}
```

17.2.3 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.

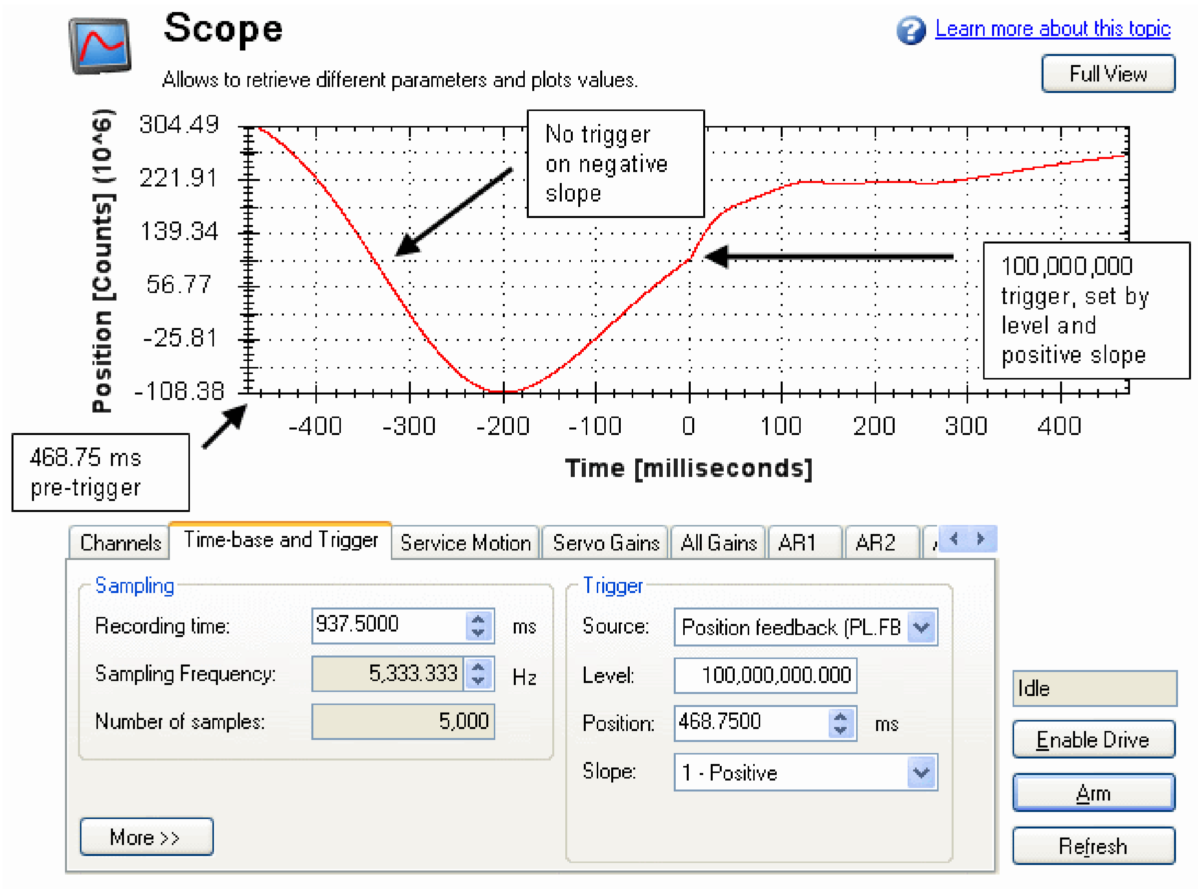


In this tab, you can set length of recording in ms and the sampling frequency in Hz. The number of samples is a calculated value displayed for reference. You can also choose the trigger source to be **Immediate** (triggers as soon as you click the **Start Recording** button) or to be one of many predefined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

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

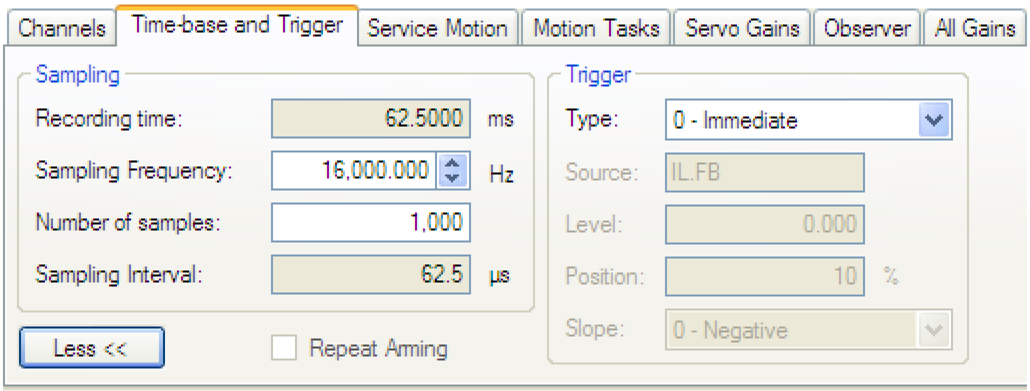
- Level sets the value of the source that triggers the recording to start.
- Position sets the amount of time that the scope displays before the trigger occurred.
- Slope sets whether the source data must pass the level value in a positive or negative direction.

An example of triggering is shown below:



17.2.3.1 Scope Time-base and Trigger, More View

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



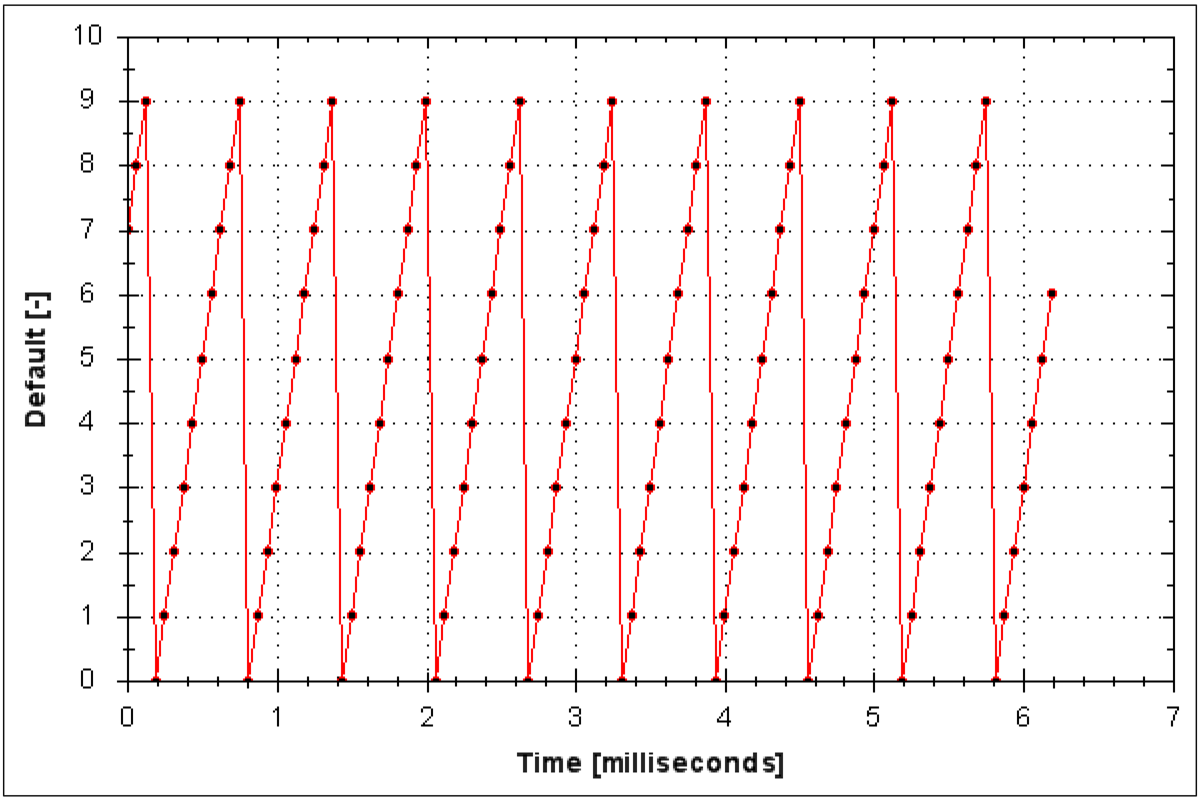
In the **Sampling** area of this view, you can specify the recording length by entering a sampling frequency and a number of samples. Here, the recording time is a calculated value displayed for reference.

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.



17.2.3.2 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).WorkBench is constantly sending telnet commands, so this is not typically used in a WorkBench 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.

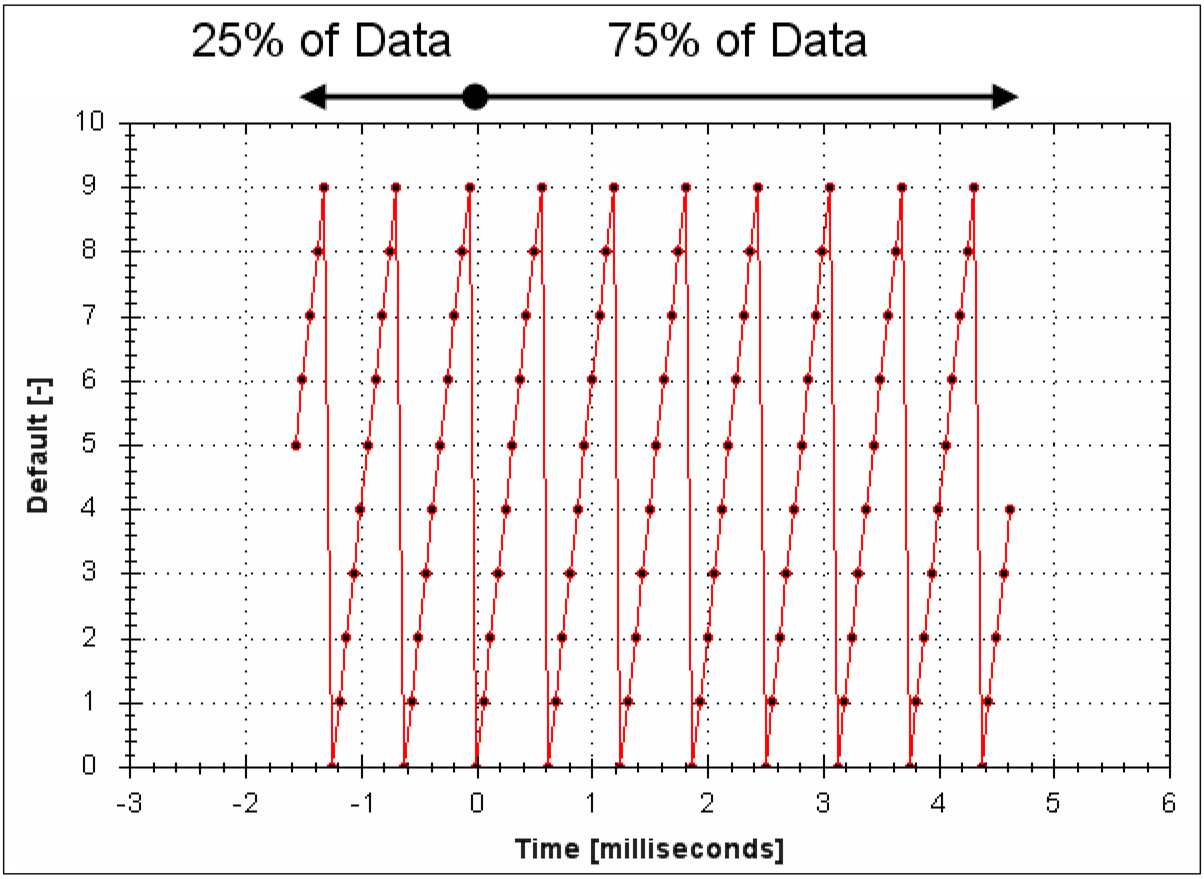
17.2.3.3 Trigger Position

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 WorkBench 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).



17.2.3.4 Trigger Value

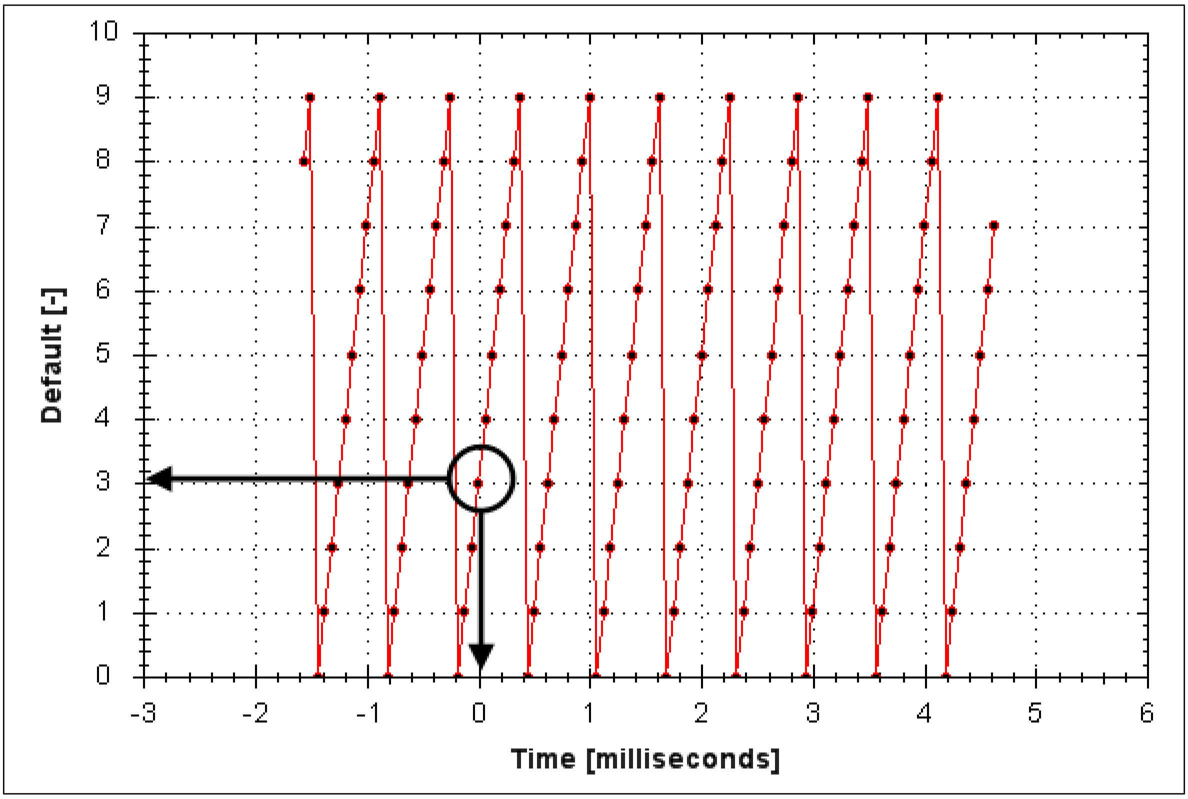
The trigger value (REC.TRIGVAL) specifies a target value that should trigger the recording to start. The trigger value is used in trigger type Parameter / On Next Signal only.

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.



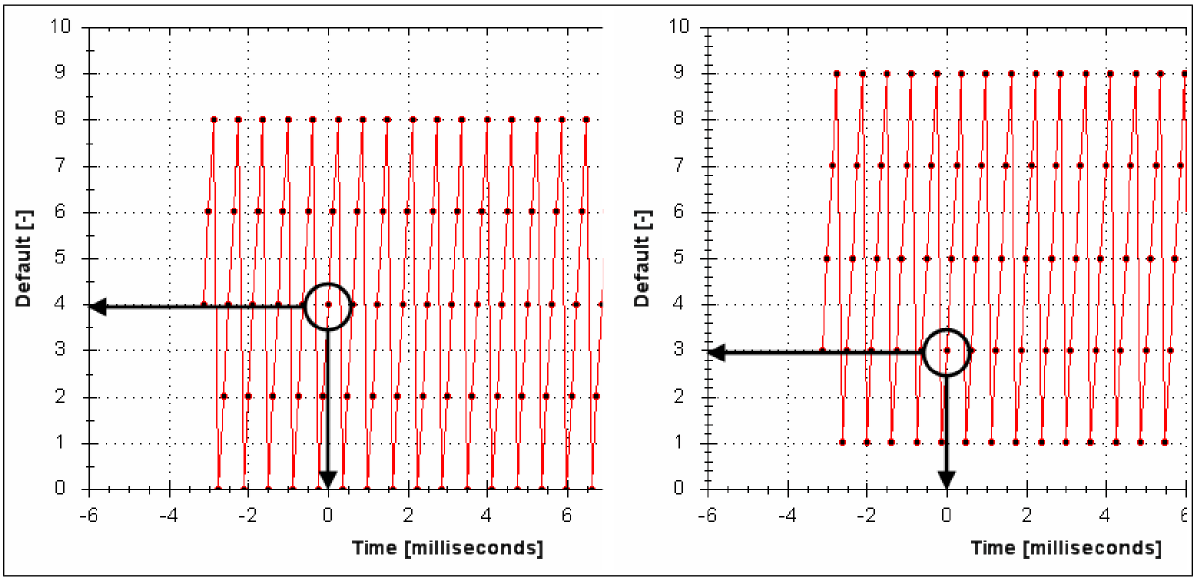
When the trigger slope is negative, the trigger value will trigger when:

- 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.

17.2.3.5 Effects of Recorder Gap

When the recording rate is less than16,000 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.

17.2.3.6 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:

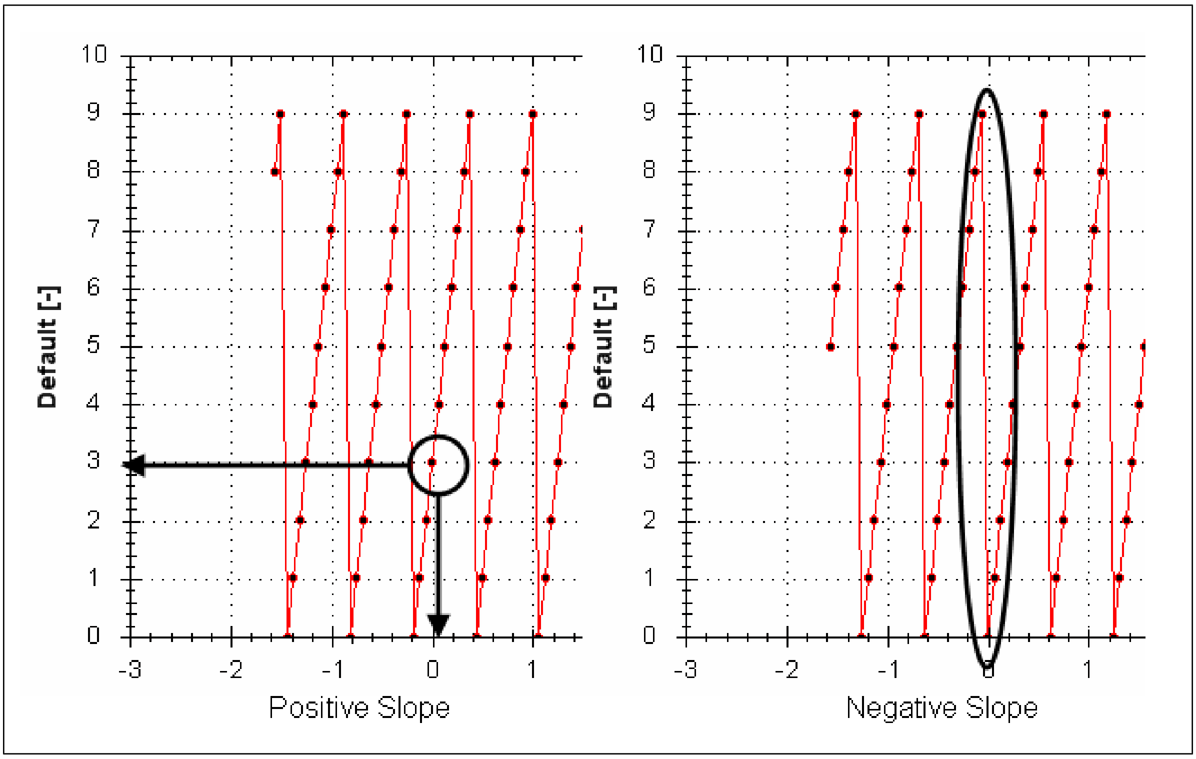
- A positive slope will trigger when the trigger source is 1
- A negative slope will trigger when the trigger source is 0

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

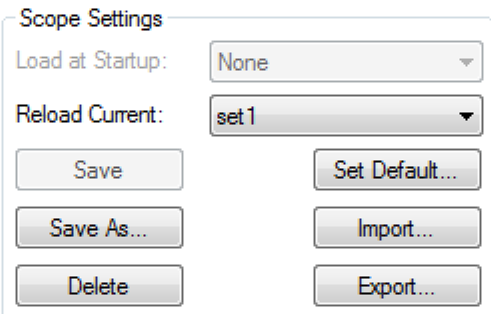
The “On Next Signal” trigger type allows you to specify if the recorder should trigger when the signal crosses the trigger level in the positive or negative direction. The signal only needs to reach the trigger level; it does not need to pass the trigger level.

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.



17.3 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 WorkBench project file (default.wbproj) and settings are common to all the drives in WorkBench.

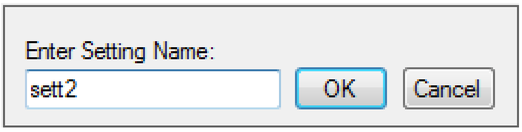


17.3.1 Load a setting (preset) to Scope screen

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.

17.3.2 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.

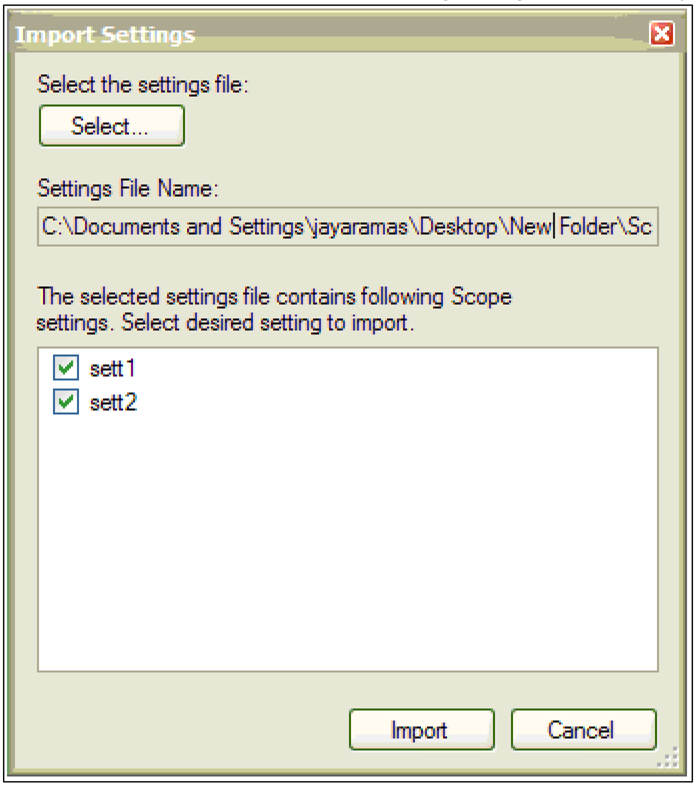
17.3.3 Save or delete preset

Save saves any modification to the open preset. **Delete** deletes the open preset.

17.3.4 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.

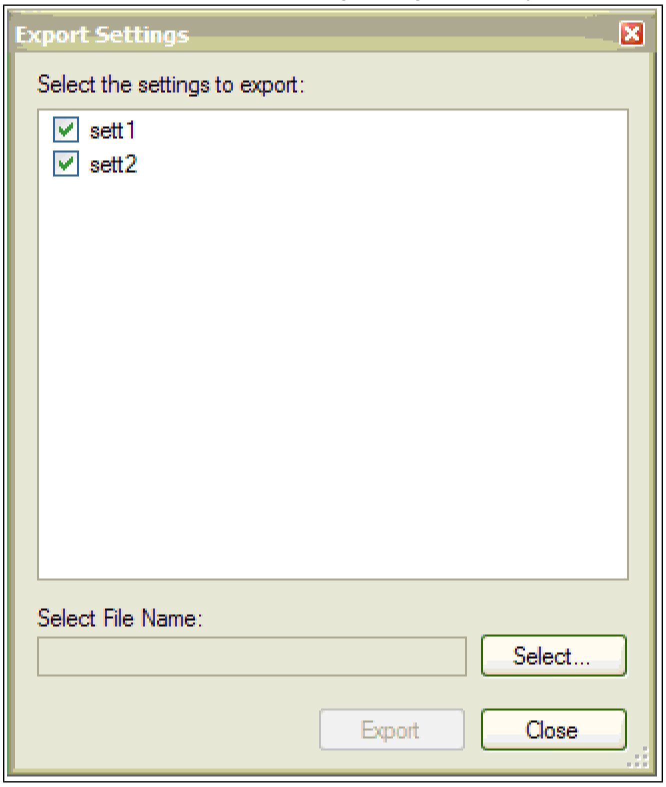


2. Select the settings file by clicking “Select…” button.
3. All the scope presets will be displayed contained in the selected settings file.
4. Select/Deselect the presets and then click on Import.
5. If preset name already exists in application the confirmation message will be shown to user to replace it or to ignore.

17.3.5 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.

17.3.6 Scope axis scaling and zooming

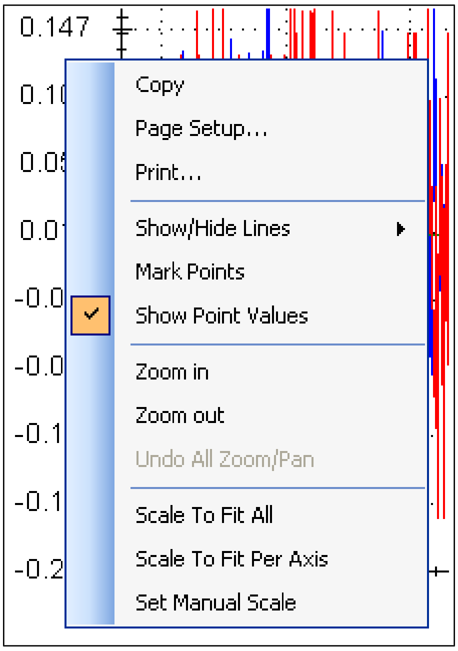
The scope provides two mechanisms for determining how you view the data:

- 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.

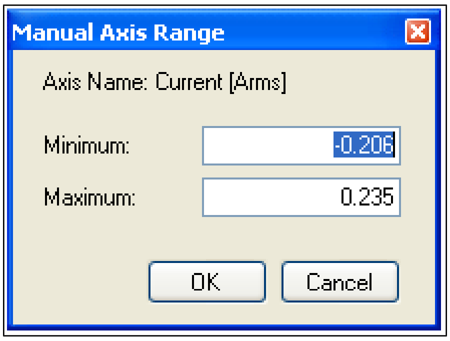


The zoom functionality allows you to navigate in a portion of the graphic. When you reset the zoom, the initial scales are shown.

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 reseted when exiting WorkBench and should be explicitly set at next startup.

17.3.7 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.



17.3.8 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 (pg 411)

18 Using Parameters and the Terminal Screen

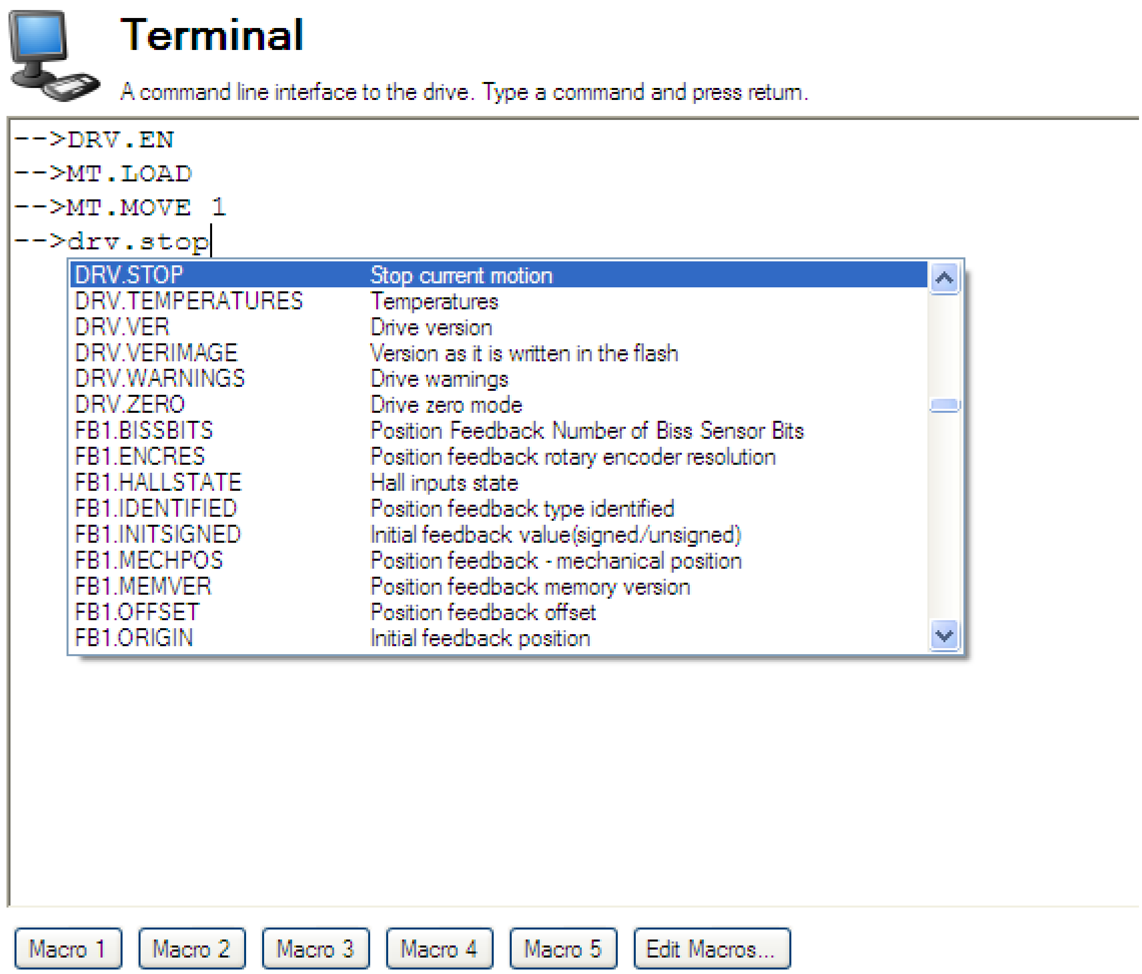
18.1 Terminal

18.1.1 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, WorkBench 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.

18.1.2 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:



The terminal supports the following keyboard shortcuts:

Keyboard Shortcut	Description
F2	Executes the last command.
Up Arrow	Gets the previous command from the command history.
Down Arrow	Gets the next command from the command history.

Keyboard Shortcut	Description
CTRL+J	Shows the list of commands that the drive supports.
CTRL+L	Opens the Parameter Load/Save view.
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.

18.1.3 Macros

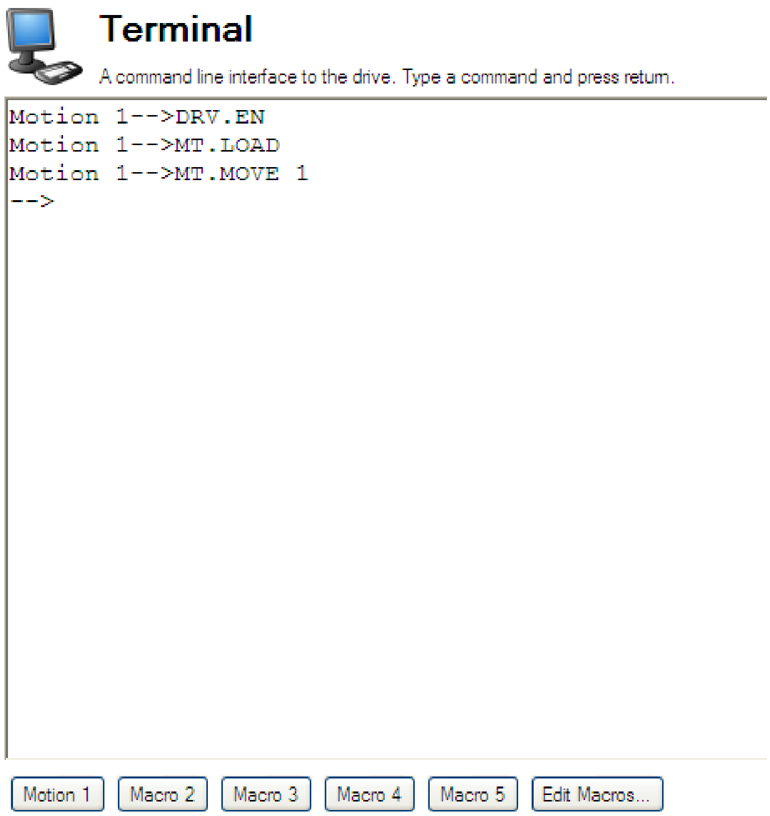
Macros are a short sequence of instructions that can be sent to the drive using a single button. You can create up to five macros within the terminal mode using the macro editor. Each of the macros can be given a name and will appear as a button below the Terminal screen area. When you select the button, 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.

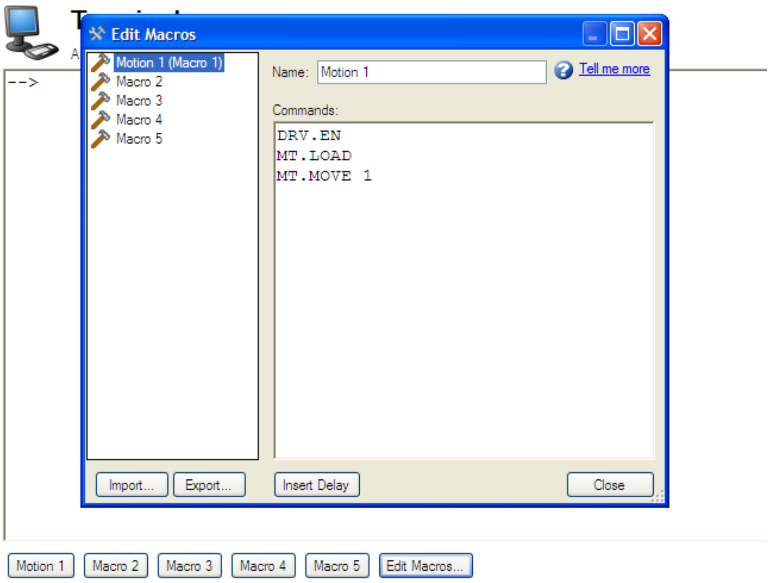
Creating a Macro from Terminal commands

You can create a macro from a sequence of terminal commands as follows:

1. Enter the sequence of commands.



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.

Macro Editor

The macro editor allows you to create and modify up to five macros.WorkBench automatically saves the macros that you create. If you exit WorkBench and then start WorkBench 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.

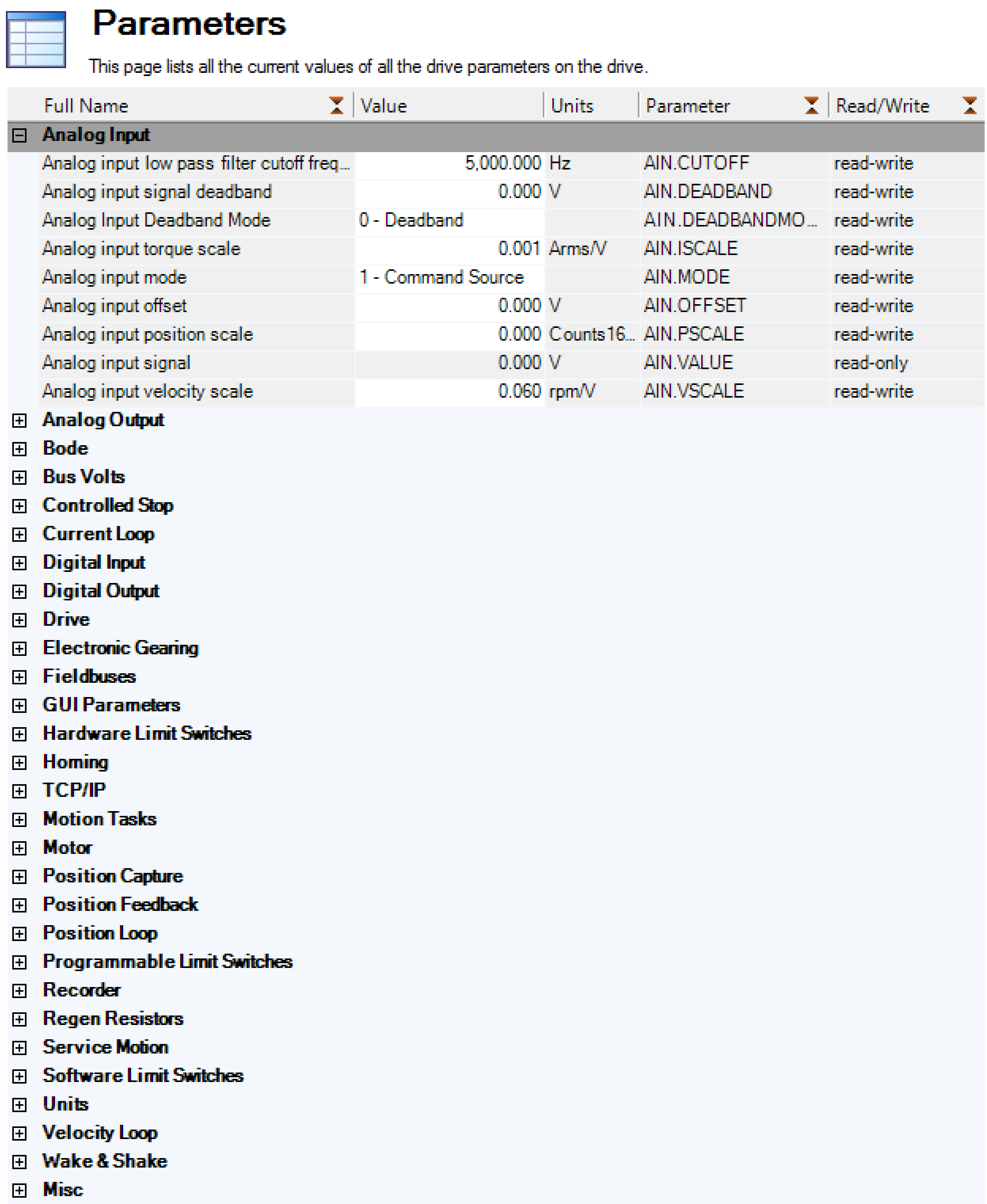
Button or Dialog Box	Description
Name	Allows you to give each macro a unique name. This name will be shown on the button in the terminal view.
Commands	Displays the commands that are sent to the drive when you use this macro. The response from the drive is displayed in the terminal.
Import	Overwrites the selected macro with the contents of a macro file exported from another computer.
Export	Sends the selected macro to a text file to that you can import into WorkBench running on another computer.
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 #delay 5000 will pause the execution of the macro for 5000 milliseconds.
Close	Closes this window and returns to the terminal view.

18.2 Viewing Parameters

You can view and edit parameters in the **Parameters** screen. You can view and write parameters in the **Terminal** screen.

18.3 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.
More/Less	Adds two more columns to the table with the range and default values for each parameter.

18.4 Parameter Load/Save

To copy a configuration (all the drive parameters that are stored in the drive's nonvolatile memory) from one drive to another, click the **Save to File** button on the first drive and then use the **Load from File** on the second drive. If you wish the second drive to keep these new parameters after the drive is turned off, then you can save the parameters to the nonvolatile memory with **Save To Drive**.

Button or Dialog Box	Description
Save To File	Copies all the drive parameters and creates a file.
Load From Drive	Sets all the drive parameters within the drive to the values contained within the file you select.
Save to Drive	Saves all drive parameters into the nonvolatile memory of the drive. Each time the drive powers on, it will start with these saved parameters.
Default	Returns all the drive parameters back to their default values. Any changes you have made are lost.

18.5 Parameter Comparer

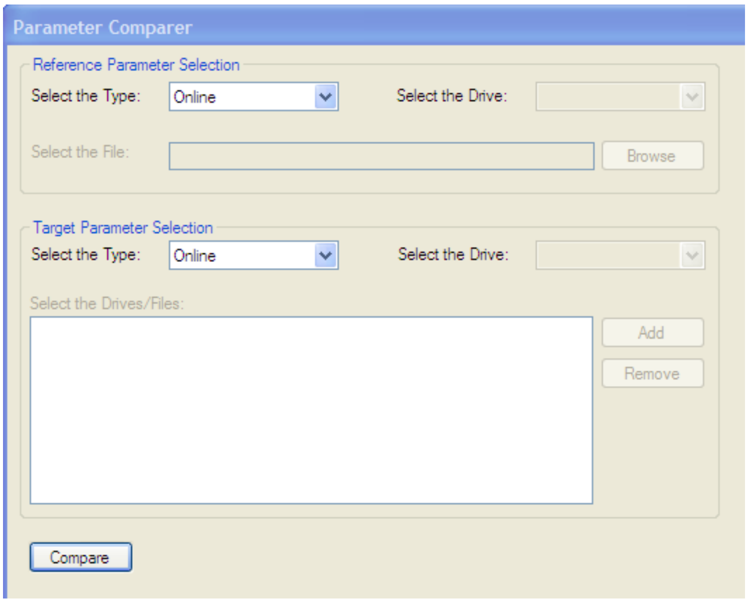
The **Parameter Comparer** tool is used to compare parameters from different drives and to compare motion tasks of different drives. You can select this tool from the menu bar, **Tools**> **Parameter Comparer**.

Parameters can be compared between the following:

- Drives (Online / Offline)
- Drives and Files
- File and Drives
- File and Files

The **Parameter Comparer** screen includes the three sections shown below:

- Reference Parameter Selection
- Target Parameter Selection
- Display Comparison



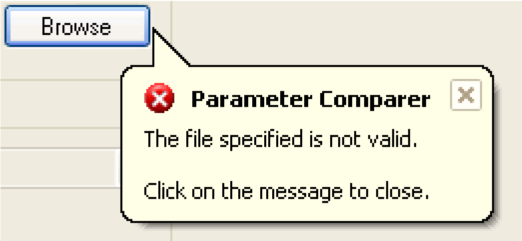
18.5.1 Reference Parameter Selection

You can select the drive type (Online/Offline) and file to compare in the Reference Parameter Selection area.

When a drive type (Online/Offline) is selected , the drives connected to the WorkBench are displayed in the **Select the drive** box. By default, the first drive is selected. If no drives are connected, then the **Select the drive**box will be in disabled state.

If **File Selection** is chosen from the **Select the Type** box, then the Select the drive box is disabled and the **Select the File** box and **Browse** button become enabled for the user to browse the parameter file (*.AKD).

A warning message is displayed if you select an incorrect file, as shown below:

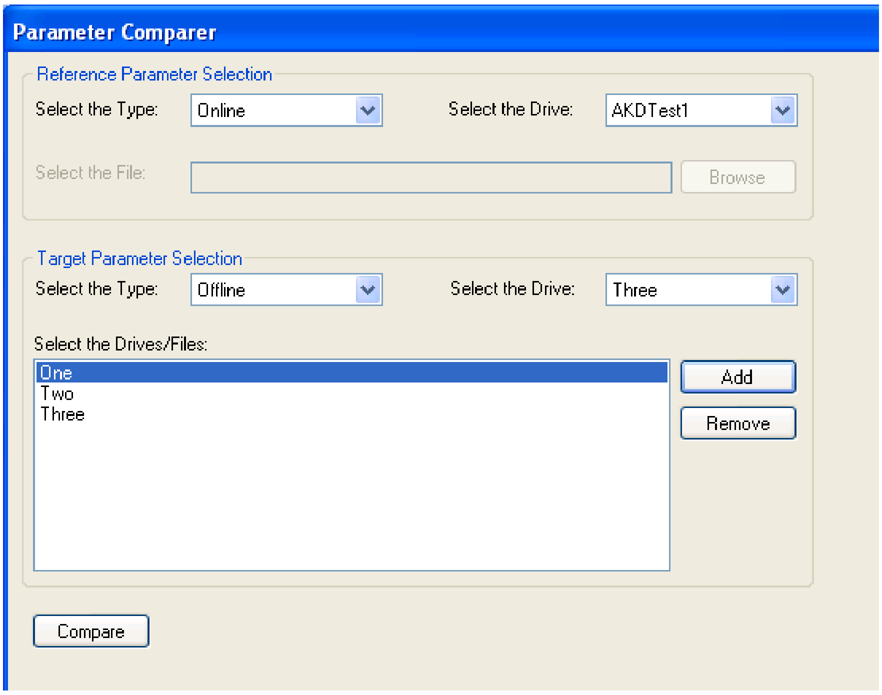


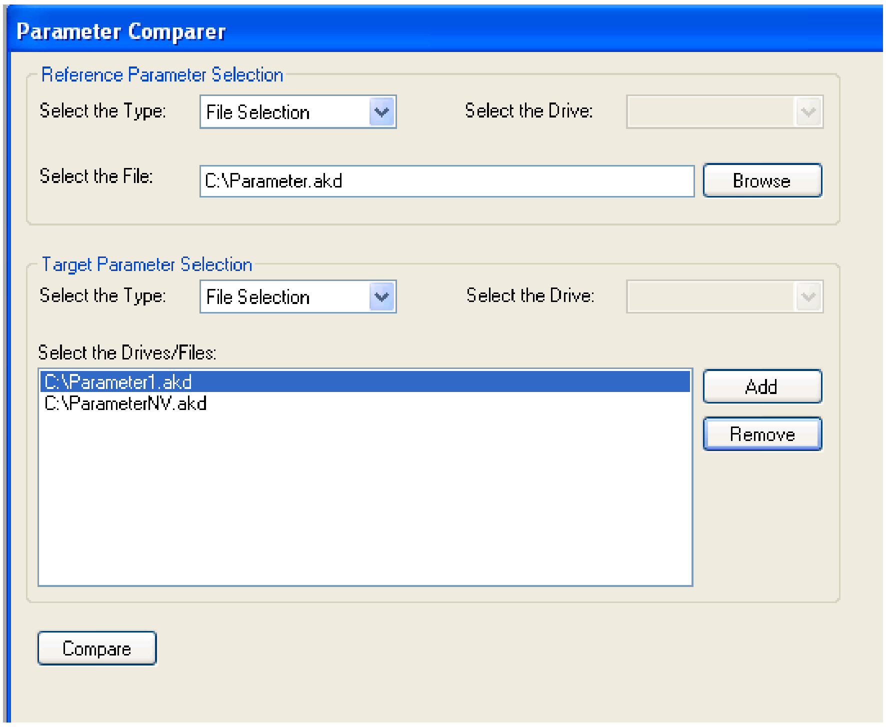
18.5.2 Target Parameter Selection

Target Parameter Selection includes the following:

- Select the Type box to select the type (drive (online/offline) and File).
- **Select the Drive** box to select the drives connected to the WorkBench.
- **Select the Drives/Files** box to load files or drives which are to be compared.
- **Add** button to add Drives/Files selected to the list box.
- **Remove** button to remove the selected item from the list box.

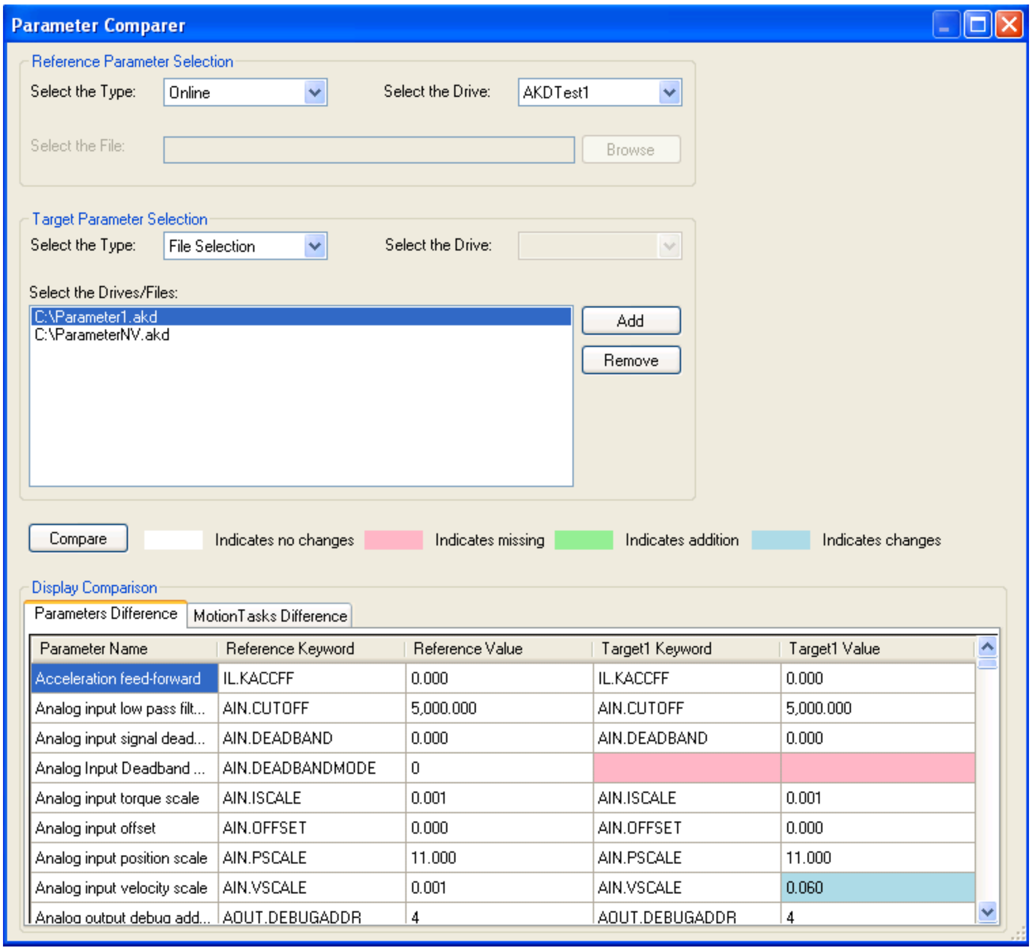
Many target drives/files parameters can be compared with one reference drive/file parameters as shown below.





18.5.3 Display the comparison

Clicking **Compare** displays the comparison shown below:



Display Comparison contains two sections:

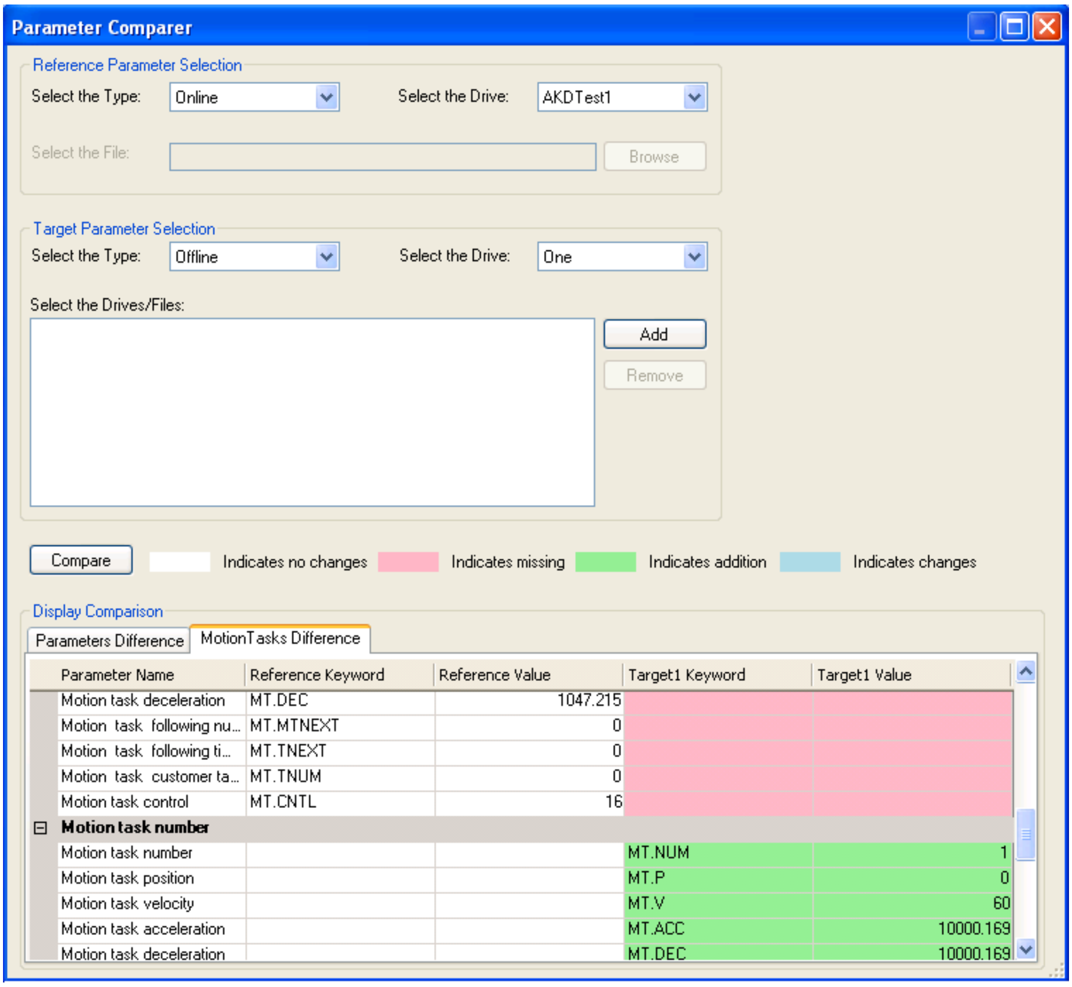
- Parameters Difference
- Motion Tasks Difference

Each of these sections displays:

- Parameter name
- Reference keyword
- Reference value
- Target keyword
- Target value

18.5.4 Motion Task Comparison

Motion Task Comparison is displayed as shown below:



A separate section is shown for each motion task.

Motion Tasks Difference and **Parameters Difference**are color coded as shown in the table below.

	Keyword exists in the reference file /drive and does not exists in the Target files/drives.
	Parameter does not exist in reference file/drive and exists in Target files/drives.
	Keyword exists in both reference and target files/drives, and the value or the parameter key-word changes.
	Reference and target parameter do not vary.

If you click the **Compare** button without selecting the drive or file, you will receive a message prompting you to make a selection.

This page intentionally left blank.

19 Faults and Warnings

19.1 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 theAKDWorkBench Fault Screen or read the status of DRV.FAULTS through the controller or HMI for the entire list of faults.

Once the fault cause is remedied, you can clear the fault using any one of the following methods:

For more detailed information on clearing drive faults, see Clearing 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 type mismatch.	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	Non-volatile memory stamp invalid.	Non-volatile memory stamp is corrupted or invalid.	Reset the drive to default memory values using Parameter Load in WorkBench.

Fault	Message/Warning	Cause	Remedy
F106	Non-volatile memory data	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 WorkBench.
n107	Positive switch limit exceeded.	Positive software position limit is exceeded.	Move the load away from the limits.
n108	Negative switch limit exceeded.	Negative software 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	Synchronization 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	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 stop 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 an integer.	Ratio of motor poles to feedback poles must be a whole number.	Change to a compatible feedback device.
F129	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).
F133	Fault number changed to to F138. See F138 for details.		
F134	Secondary feedback illegal state.	Feedback signals were detected in an illegal combination.	Check X9 connection.

Fault	Message/Warning	Cause	Remedy
F135 n135	Homing is needed.	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 Overshot 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.	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 stetting of VBUS.HALFVOLT.

Fault	Message/Warning	Cause	Remedy
n151	Not enough distance to move; motion exception.	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. 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.	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.
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.

Fault	Message/Warning	Cause	Remedy
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 to F232	EEPROM failure detected	EEPROM failure detected	Restart drive. If issue persists, exchange drive.
F234-F237 n234-n237	Temperature sensor high.	High temperature limit reached.	Check cabinet ventilation system.

Fault	Message/Warning	Cause	Remedy
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	Bus voltage exceed allowed 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.
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.
F303	Runaway.	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 motor.brake = 100.
F306	Brake short circuit.	Motor brake short circuit.	Check cabling and general functionality.
F307	Brake closed during enable state.	Motor brake closed unexpectedly.	Check cabling and general functionality.
F308	Voltage exceeds motor rating.	Drive bus voltage exceeds the motor's defined voltage rating.	Make sure that the motor fits the driving rating.

Fault	Message/Warning	Cause	Remedy
F309	Motor I2t load. reduce load	Motor I2t load (IL.MI2T) has exceeded the warning threshold IL.MI2TWTHRESH. This warning can only be generated in the case that the motor protection mode IL.MIMODE 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
F404	Hall error.	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 multicycle 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 Communications 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.

Fault	Message/Warning	Cause	Remedy
F421	SFD position sensor fault	Sensor or sensor wiring failure inside motor.	Try resetting the fault. If it reappears return motor for repair.
F423	NV Failure, Multiturn Overflow.	The position saved in memory is corrupted.	Home axis or disable multiturn overflow. If the fault persists, send the drive for repair.
F424	Resolver amplitude low.	Resolver signal amplitude is below minimum level.	Check primary feedback (X10 connection).
F425	Resolver amplitude high.	Resolver signal amplitude is above maximum level.	Check primary feedback (X10 connection).
F426	Resolver error.	Resolver excitation fault.	Check primary feedback (X10 connection).
F427	Analog low.	Analog signal amplitude low.	Check primary feedback (X10 connection).
F428	Analog high.	Analog signal amplitude high.	Check primary feedback (X10 connection).
F429	Incremental low.	Incremental encoder signal amplitude is below minimum level.	Check primary feedback (X10 connection).
F430	Incremental high.	Incremental encoder signal amplitude is above maximum level.	Check primary feedback (X10 connection).
F432	Communication fault.	General communication problem with secondary feedback.	Check secondary feedback (X10 connection).
F438 n439	Following error (numeric)	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 (user).	Motor did not follow command values. Motor exceeded maximum allowed position following error (user).	Check feedback commutation setup and tuning parameters.
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 is not powered. The n451 warning is generated if the AKD is powered. This fault can be inhibited with FAULT451.ACTION.	Check or replace the external battery.

Fault	Message/Warning	Cause	Remedy
F452	Multiturn overflow not supported with this feedback.	Non-multiturn feedback is connected while FB1.PMTSAVEEN is active.	Connect multiturn feedback to the drive or disable multiturn overflow.
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.
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.
F467	Feedback fault (See FB1.FAULTS for details).	The Hiperface DSL device malfunctioned.	Check FB1.FAULTS for detailed fault information.
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.

Fault	Message/Warning	Cause	Remedy
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 suported for vertical/overhung loads.
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	Fieldbus command velocity too high.	Fieldbus command velocity exceeds VL.LIMITP.	Lower fieldbus command trajectory, or increase the value of VL.LIMITP.
F481	Fieldbus command velocity too low.	Fieldbus command velocity exceeds VL.LIMITN.	Increase fieldbus command trajectory, or decrease the value of VL.LIMITN.
F482	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	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	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	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	Motor velocity exceeds EMU-speed.	Motor velocity exceeds the maximum speed the emulated encoder output can generate.	Reduce value of DRV.EMU-EPULSEIDTH.
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.

Fault	Message/Warning	Cause	Remedy
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.
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: • 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 overload.	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.
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.

Fault	Message/Warning	Cause	Remedy
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.	Current sensor short circuit.	Restart drive. If issue persists, contact technical support.
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.
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 DRV.SETUPREQLIST 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 (MOTOR.AUTOSET 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 MOTOR.AUTOSET 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 DRV.CLRFAULTS command. If this attempt is not successful, then set MOTOR.AUTOSET 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 fail.	The power-board temperature sensor indicates more than 85 °C.	Reduce the load of the drive or ensure better cooling.

Fault	Message/Warning	Cause	Remedy
F560	Regen near capacity, could not prevent over voltage.	An F501 Bus Over Voltage has occured 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.
F601	Modbus data rate is too high.	Modbus controller data rate is too high.	Reduce data rate.
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 andDRV.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.
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 while axis should disable	Motor did not stop in the timeout defined.	Change timeout value, change stop parameters, improve tuning.

19.2 Additional Fault Messages AKD-T

AKD BASIC runtime faults are displayed in the two-digits 7-segment display of the drive:

The two digits LED display indicates the fault code.

The additional runtime fault messages for AKD-T are coded with numbers starting from F801. All faults activate the fault relais and can be cleared with DRV.CLRFAULTS.

NOTE More information about fault messages and clearing faults can be found in the WorkBench online help.

Remedy for all errors: Clear Fault, fix user program, recompile, download and attempt to run the program again.

Error	Description	Cause
F801	Divide by zero.	User Program attempted to divide by zero.
F802	Stack Overflow.	User Program contains an infinite recursion or incorrectly array.
F803	Insufficient Memory.	User program creates an excessive demand for memory.
F804	No interrupt handler defined.	User program is missing an interrupt service routine, but an interrupt is called.
F805	Interrupt error.	User program contains an error in an interrupt routine.
F806	Max string length exceeded.	User program attempted to use a string exceeding 255 characters.
F807	String overflow.	User program has a exception causing excessive string usage.
F808	Array out of bounds.	User program exception caused an array to exceed its bounds.
F809	Feature not supported.	User program contains a feature that the current firmware version does not support.
F810	Internal firmware/hardware error.	User program attempted to perform an action that causes a firmware or hardware error.
F812	Parameter not supported.	User program calls a parameter that is not supported by the firmware.
F813	Parameter access error.	User program contains a parameter access error.
F814	Data not found.	User program attempted writing an invalid recorder parameter.
F815	Data invalid.	User program attempted executing an invalid command.
F816	Data too high.	User program contains a parameter that is above the accepted range.
F817	Data too low.	User program contains a parameter that is below the accepted range.
F818	Param type out of range.	User program attempted to write a value which was out of a range.
F819	Data not divisible by 2.	User program executed a function that requires it to be divisible by two.
F820	Invalid position modulo setting.	User program contains an incorrectly configured modulo setting.
F821	Cannot read from command.	User program attempted to perform a read of parameter that is a command or statement.
F823	Enable Drive first.	User program is attempting to execute motion that requires the drive to be enabled.

Error	Description	Cause
F824	DRV.OPMODE must be 2 (position).	User program is attempting to execute motion that requires the drive to be in program mode.
F825	DRV.CMDSOURCE must be 5 (program).	User program is attempting to execute motion that requires the drive to be in position mode.
F826	Cannot execute during a move.	User program is attempting an invalid execution during a move.
F827	Writing to read-only parameter.	User program attempted writing to a read-only parameter.
F828	Disable Drive first.	User program tries to execute a function that requires a disabled drive.

19.3 SD Card Errors

Fault	Message/Warning	Cause	Remedy
E0011	Drive enabled! Disable drive first.	Cannot perform SD.SAVE or SD.LOAD while drive is enabled.	Disable drive.
E0064	Cannot execute when program is running.	Program is running.	Stop program.
E0066	Command is password protected	The BASIC program in drive is password protected.	De-activate password protection from program.
E0082	SD Card is not inserted. With I/O option card only.	No SD card inserted or SD card inserted with wrong orientation.	Insert SD card with correct orientation.
E0083	SD Card is write protected. With I/O option card only.	SD card protection clip in wrong position.	Remove write protection from SD card.
E0084	SD Card hardware not installed.	No I/O option board installed or SD card device faulty.	-
E0095	File not found on SD Card. With I/O option card only.	SD card damaged or filename has been changed manually or deleted.	-
E0096	File error trying to access the SD Card. With I/O option card only.	File on SD card can't be read.	-
E0097	File system error accessing the SD card. With I/O option card only.	File system on SD card can't be read.	Use supported SD cards only (see SD Card Slot in Installation Manual)
E0098	A parameter could not be set in the drive. With I/O option card only.	-	-
E0099	There was an error writing to a file on the SD Card. With I/O option card only.	-	-
E0100	SD Card read/write in progress. With I/O option card only.	-	Wait until read/write process is done.
E0101	There was an error accessing the BASIC binary file. With I/O option card only.	Basic program file couldn't be read.	-
E0102	BASIC program missing or invalid.	SD.SAVE and SD.LOAD are not possible if a BASIC program does not exist.	

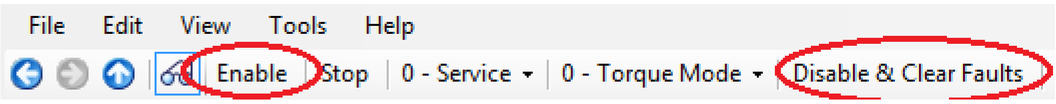
19.4 Clearing Faults

You can clear drive faults in several ways:

1. using the WorkBench toolbar
2. using the WorkBench terminal screen
3. using the CANopen interface
4. using an external digital signal

In the first three methods, 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). In the fourth method, 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 WorkBench 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 WorkBench terminal screen, enter DRV.CLRFAULTS, then enter DRV.EN to enable the drive.
3. Using the CANopen interface (Object 6040h: Controlword DS402). The ASCII control commands are built up from the logical combination of the bits in the control word and external signals (enable output stage). You can configure the **Reset Fault** bits in the control word to clear the fault. See 1 Object 6040h: Control word (DS402) in the CANopen manual for details on clearing faults using this method. After the fault is cleared, you must enable the drive.
4. 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 WorkBench terminal screen, enter DINx.MODE 1, with x set to the corresponding digital input number. See the Installation Manual (Digital Inputs (X7/X8) (pg 108) 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).

19.5 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 WorkBench terminal screen to find valid parameters and commands.	General.

Error Message	Meaning	Remedy	Occurrence
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.
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 WorkBench 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 WorkBench 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 WorkBench 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 withing 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 Message	Meaning	Remedy	Occurrence
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: 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 Message	Meaning	Remedy	Occurrence
Error: Invalid motor/feedback poles ratio.			
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 Message	Meaning	Remedy	Occurrence
Error: Iu/Iv offset calculation not finished.			
Error: Buffer overflow.			
Error: Cannot save to EEPROM while inrush relay is closed.			
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.			

19.6 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 from 0000h to 7FFFh are defined in the communication or drive profile. Error numbers from FF00h to FFFFh have manufacturer-specific definitions. The following table describes the various error codes:

Error Code	Fault/ Warning Code	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^2T.

Error Code	Fault/ Warn-ing Code	Description
0x3282	F519	Regen short circuit.
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.
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
0x5510	F201	Internal RAM failed.
0x5530	F105	Hardware memory, non-volatile memory stamp invalid
0x5580	F106	Hardware memory, non-volatile memory data
0x5581	F202	Hardware memory, external Ram for resident firmware failed
0x5582	F203	Hardware memory, code integrity failed for resident firmware
0x5583	F102	Hardware memory, resident firmware failed
0x5584	F103	Hardware memory, resident FPGA failed
0x5585	F104	Hardware memory, operational FPGA failed
0x6380	F532	Drive motor parameters setup incomplete.
0x7180	F301	Motor overheat

Error Code	Fault/ Warn-ing Code	Description
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.
0x7303	F426	Resolver error
0x7305	F417	Broken wire in primary feedback
0x7380	F402	Feedback 1 Analog signal amplitudefault
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
0x738B	F415	Feedback 1 SFD mult. corrupt position
0x738C	F416	Feedback 1 SFD Transfer incomplete
0x738D	F418	Feedback 1 power supply fault
0x738E	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.
0x73A0	F424	Feedback 1 Resolver amplitude low
0x73C0	F473	Wake and Shake. Insufficient movement
0x73C1	F475	Wake and Shake. Excessive movement.
0x73C2	F476	Wake and Shake. Fine-coarse delta too large.
0x73C3	F478	Wake and Shake. Overspeed.
0x73C4	F479	Wake and Shake. Loop angle delta too large.
0x73C5	F482	Commutation not initialized
0x73C6	F483	Motor U phase missing.
0x73C7	F484	Motor V phase missing.
0x73C8	F485	Motor W phase missing.
0x73C9	n478	Warning: Wake and Shake. Overspeed.
0x73CA	n479	Warning: Wake and Shake. Loop angle delta too large.
0x8130	F129	Life Guard Error or Heartbeat Error
0x8180	n702	Warning: Fieldbus communication lost.
0x8280	F601	Modbus data rate is too high.
0x8311	F304	Motor foldback.

Error Code	Fault/ Warn-ing Code	Description
0x8331	F524	Drive foldback.
0x8380	n524	Warning: Drive foldback
0x8381	n304	Warning: Motor foldback
0x8382	n309	Warning: Motor I²t load.
0x8480	F302	Over speed
0x8482	F480	Fieldbus command velocity too high
0x8481	F703	Emergency timeout occurred while axis should disable
0x8483	F481	Fieldbus command velocity too low.
0x8580	F107	Software limit switch, positive
0x8581	F108	Software limit switch, negative
0x8582	n107	Warning: Positive software position limit is exceeded.
0x8583	n108	Warning: Negative software position limit is exceeded.
0x8584	n704	Warning: PVT buffer overflow
0x8585	n705	Warning: PVT buffer underflow
0x8586	n127	Warning: Scale factor of PVT velocity command over range.
0x8611	F439	Following error (user)
0x8684	n123	Warning: Invalid motion task
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.
0x86B0	F438	Following error (numeric)
0x8780	F125	Fieldbus synchronization lost
0x8781	n125	Warning: Fieldbus synchronization lost
0x8AF0	n137	Warning: Homing and feedback mismatch
0x8AF1	n140	Warning: VBUS.HALFVOLT has changed.
0xFF01	F702	Fieldbus communication lost

Error Code	Fault/ Warn-ing Code	Description
0xFF02	F529	Iu current offset limit exceeded
0xFF03	F530	Iv current offset limit exceeded
0xFF04	F521	Regen over power
0xFF05	F527	Iu current AD converter stuck
0xFF06	F528	Iv current AD converter stuck
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.

19.7 Unknown Fault

This fault message occurs when an undefined fault condition is encountered.

19.7.1 Remedies

1. Click **Clear Fault**.

This page intentionally left blank.

20 Troubleshooting the AKD

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.

Problem	Possible Causes	Remedy
HMI message: Communication fault	• wrong cable used, cable plugged into wrong position on drive or PC • wrong PC interface selected	• plug cable into the correct sockets on the drive and PC • select correct interface
Drive does not enable	• HW Enable not wired • HW or SW Enable not set	• connect HW Enable (X8 pin 4) • Apply 24V to HW Enable and select SW Enable in WorkBench / Fieldbus
Motor does not rotate	• 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	• apply ENABLE signal • set software enable • check setpoint cable • correct motor phase sequence • check brake control • check mechanism • set motor pole no. • set up feedback correctly
Motor oscillates	• gain is too high (speed controller) • feedback cable shielding broken • AGND not wired up	• reduce VL.KP (speed controller) • replace feedback cable • join AGND to CNC-GND
Drive reports following error	• Irms or Ipeak set too low • current or velocity limits apply • accel/decel ramp is too long	• verify motor/drive sizing • verify that IL.LIMITN/P,VL.LIMITN/P are not limiting the drive • reduce DRV.ACC/DRV.DEC
Motor overheating	• motor operating above its rating • motor current settings incorrect	• verify motor/drive sizing • verify motor continuous and peak current values are set correctly
Drive too soft	• Kp (speed controller) too low • Ki (speed controller) too low • filters set too high	• increase VL.KP (speed controller) • increase VL.KI (speed controller) • refer to documentation regarding reducing filtering (VL.AR*)

Problem	Possible Causes	Remedy
Drive runs roughly	• Kp (speed controller) too high • Ki (speed controller) too high • filters set too low	• reduce VL.KP (speed controller) • reduce VL.KI (speed controller) • refer to documentation regarding increasing filtering (VL.AR*)
During installation, the message “Please wait while the installer finishes determining your disk space requirements” appears and never disappears.	• MSI installer issue. • Harddisk space not sufficient	• Cancel the installation. Relaunch the installer (you may need to try several times, the problem is random). • Make sure that you have enough disk space on your hard disk (~500MB to allow Windows .NET update if necessary), if not make some space.

21 Firmware and Firmware Updates

21.1 Downloading Firmware

To download firmware, click **More** on the **AKD Overview** screen (see Drive Overview (pg 52)). The information displayed includes the current firmware version. Click **Download** to display the **Download Firmware** view and update the firmware.

When you download the firmware, you may see a **Save** dialog box; see Save On Firmware Download (pg 181) for more information about saving your drive configuration prior to a firmware download.

During the download , you cannot perform any actions with the drive.

⚠CAUTION While WorkBench is downloading firmware to your drive, do not remove the 24 V logic power. If you remove the 24 V 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 WorkBench will prompt you to reload the firmware.

During the firmware download, the Download Firmware view displays a progress bar and the following messages as the download proceeds:

1. **Reading the firmware file**. This step duration depends on where the file is physically stored.
2. **Resetting the drive**. This step takes about 10 seconds.
3. **Erasing the old firmware**. This step takes about 20 seconds.
4. **Downloading the new firmware to the drive**. The drive downloads the new firmware and then programs the new firmware into the nonvolatile memory of the drive. This step takes about 20 seconds.
5. **Resetting the drive**. This step takes about 10 seconds.

During the download process, the drive LED displays [dL]. Additional codes may appear during the download; see Display Codes (pg 31) for a description of display codes related to a firmware download.

21.1.1 Firmware Compatibility

WorkBench is compatible with all previous drive firmware versions. Any new WorkBench versions issued are compatible with older firmware versions. WorkBench forward compatibility, however, cannot be ensured. When a behavior in the firmware changes, an older WorkBench version may not function correctly with the new firmware. Kollmorgen recommends that you update WorkBench when you install new firmware.

The table below describes compatibility between WorkBench and firmware versions.

	Firmware version M_01-01-00-zzz	Firmware version M_01-02-00-zzz	Firmware version M_01-03-00-zzz	Firmware version M_01-04-00-zzz	Firmware version M_01-05-00-zzz
WorkBench Version 1.1.x.x	√	x	x	x	x
WorkBench Version 1.2.x.x	√	√	x	x	x
WorkBench Version 1.3.x.x	√	√	√	x	x
WorkBench Version 1.4.xx	√	√	√	√	x
WorkBench Version 1.5.x.x	√	√	√	√	√

Any new firmware not listed here may have compatibility issues with WorkBench. Please check for latest version of WorkBench on the Kollmorgen website:

http://www.kollmorgen.com/akd_

21.2 Forcing the drive into firmware download mode

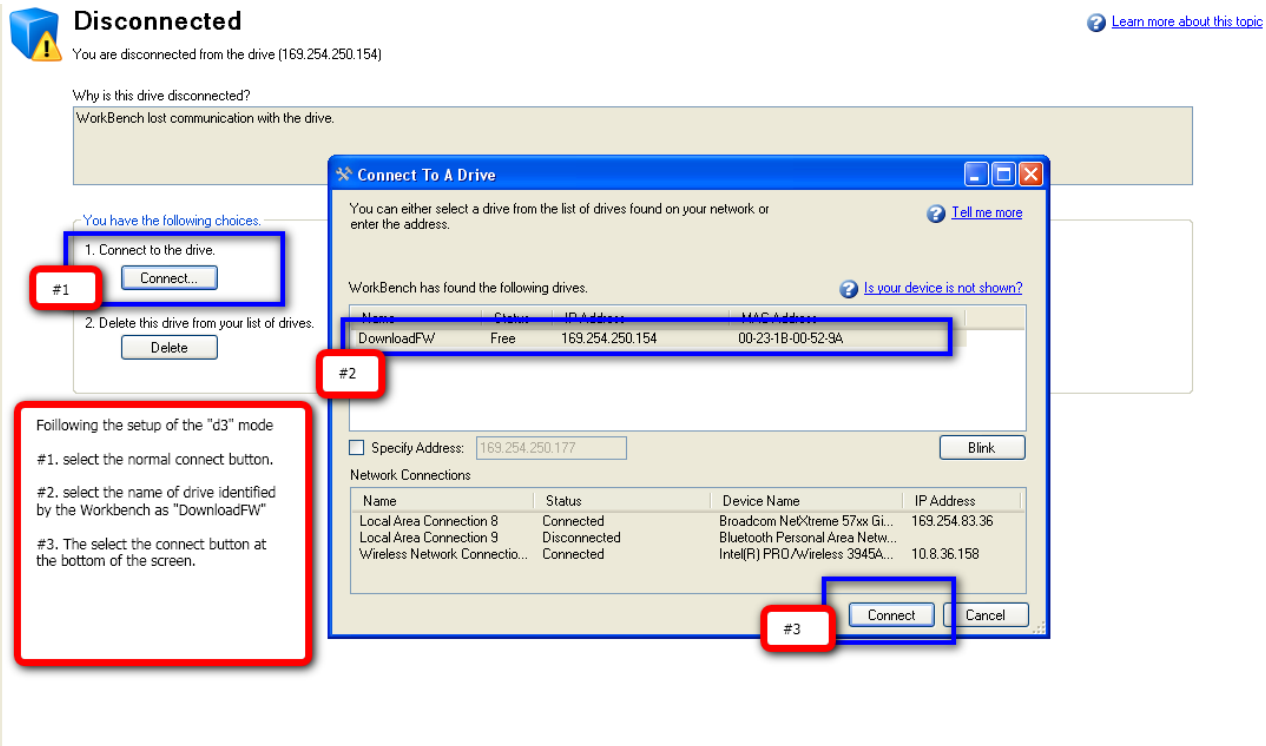
21.2.0.1 Forced download of AKD firmware.

In most cases, you can download firmware using the WorkBench without setting the hardware into a specific download state. If you are unable to connect to WorkBench for some reason, the following “forced firmware download” procedure is needed.

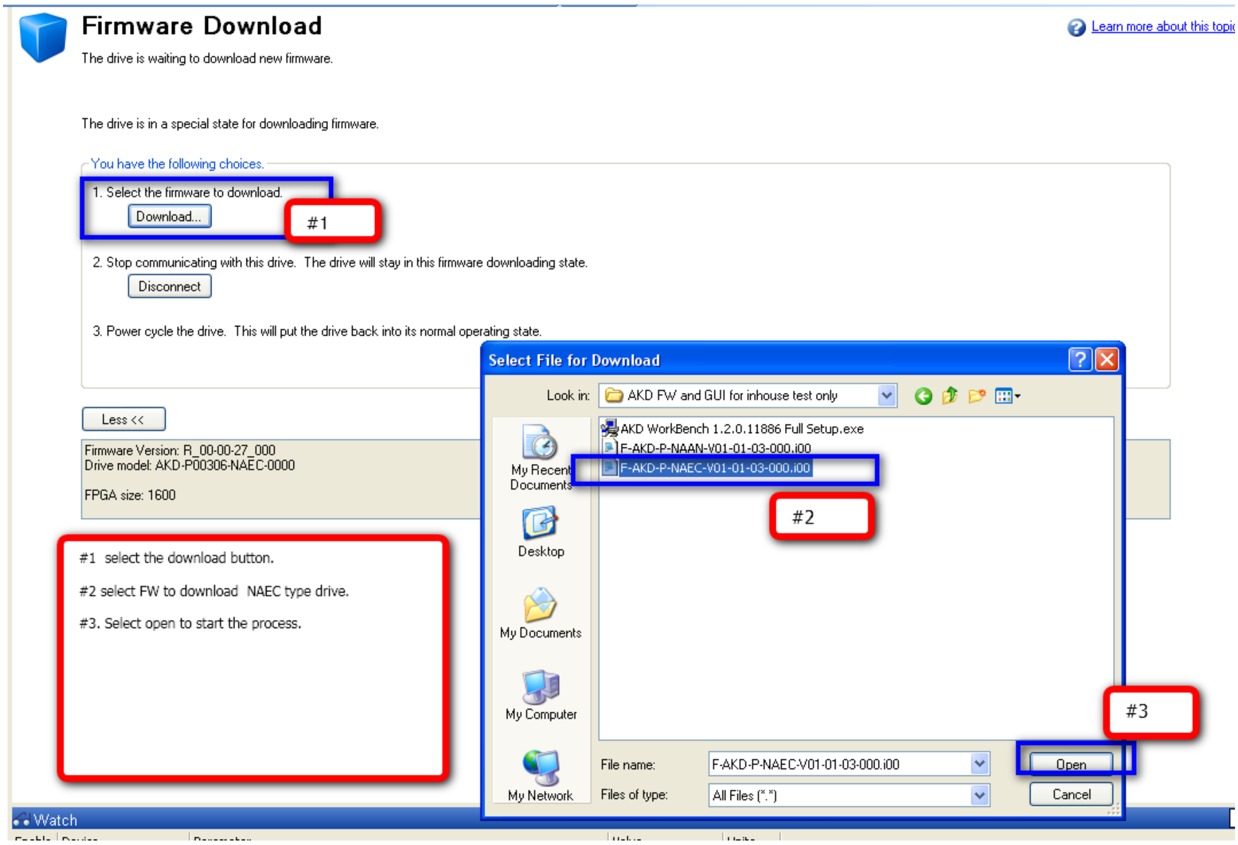
Power the AKD unit down then:

- Depress the B1 button (display of IP Address Button) while powering up the drive. Continue to depress the button until the drive display shows "d3 " (5-10 sec).
- Once the display on the drive shows "d3" , have the EtherNet cable connected to the drive Point to Point.
- The drive can now be connected to WorkBench in download mode.
- Select the firmware to be downloaded; click **Open** to start the process.

WorkBench will display the **Disconnected** view. See the screenshots below for the download connection sequence.



The **Firmware Download** view appears next. See the screenshot below for the firmware selection and download sequence.



- Next, the **Downloading Firmware** view appears and the progress bar displays the firmware download progress.
- Once the download is complete, open the **AKD Overview** screen. The **Firmware Version** box displays the new firmware version, which confirms your successful download.

21.3 Invalid Firmware

When WorkBench connects to a drive, it checks to make sure that the drive is executing a valid version of the drive firmware. If the drive is executing firmware earlier than version 1-01-00-000, then WorkBench cannot operate correctly and shows these options.

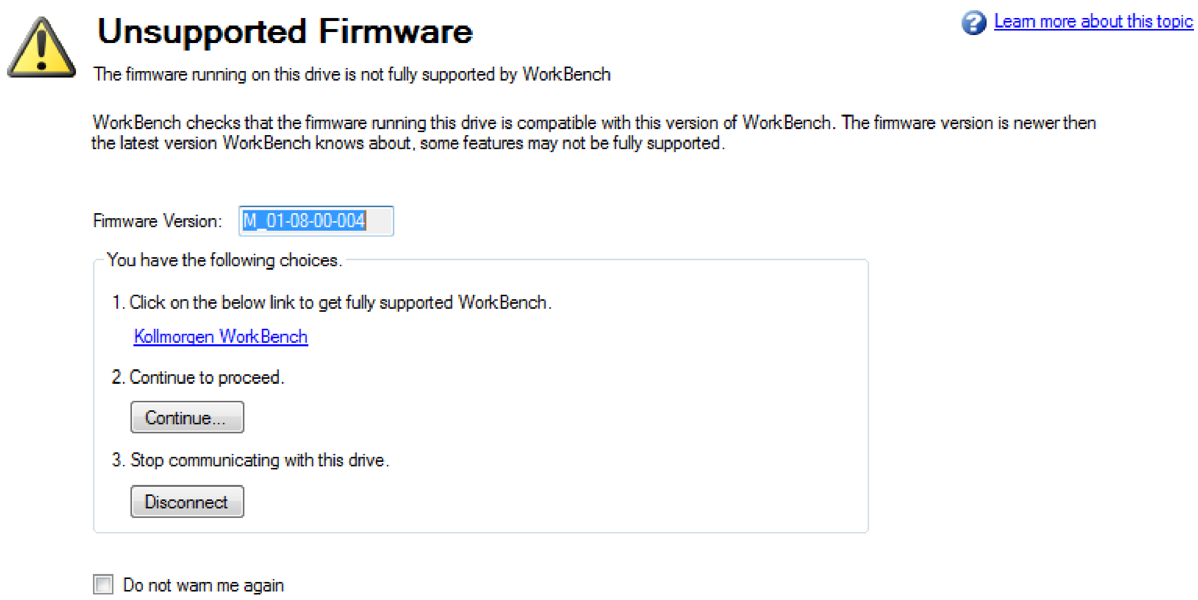
Button	Description
Download	This command allows you to select a different version of the firmware and download it to the drive.
Disconnect	This command stops all communications with the drive and return to the disconnected state.

See Also

Unsupported Firmware (pg 282) | Incompatible Firmware (pg 283)

21.4 Unsupported Firmware

When WorkBench connects to a drive, it checks to make sure that the drive is executing a supported version of the drive firmware. If the firmware version is newer thanWorkBench can recognize, the following warning is displayed.



Some features available in your firmware may not be fully supported by your WorkBench version. Choose one of the three options available on the warning screen:

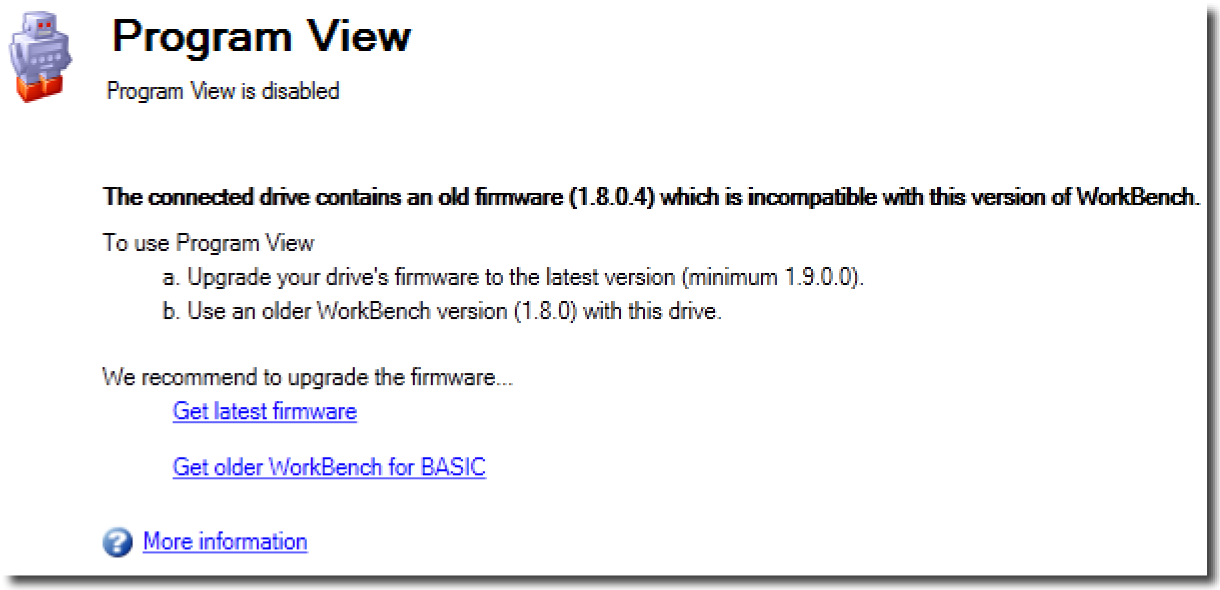
Option	Description
1	Provides link to download the latest version of WorkBench from kollmorgen.com.
2	Connects to the drive regardless of unsupported firmware.
3	Disconnects from the drive.

See Also

Invalid Firmware (pg 282) | Incompatible Firmware (pg 283)

21.5 Incompatible Firmware

If your BASIC firmware is incompatible, an incompatibility error will display in the program view.



This error is due to major WorkBench and compiler changes since your last firmware update, causing your current firmware to be incompatible with your current WorkBench version. These changes improved the product's ability to handle revision control of the language. However, the improvement also required changes in how WorkBench and firmware communicate, which caused the incompatibility. The improvement will allow the editor to recognize the firmware version being used and auto-recognize the parameters and commands inherent to that firmware. This allows better use of mis-matching WorkBench and firmware versions in the future.

See VM.PGMCOMPILERVERSION (pg 1) and VM.PGMERRORLIST (pg 1) for more information on using this new feature.

21.5.1 Resolving Incompatible Firmware

If your firmware is incompatible, we recommend upgrading the AKD BASIC's firmware to version 1.9 or later in order to take advantage of the new feature and allow seamless integration moving forward.

If the product using AKD BASIC is deep in its product lifecycle, or there are version controls in place that prevent changes to the machine device firmware, upgrading the firmware may not be suitable. In this case, downgrading to an older version of WorkBench (version 1.8.0.32358 or earlier) is recommended.

NOTE It is not necessary to uninstall your current WorkBench version in order to install an older version. Multiple versions of WorkBench can be installed simultaneously on one PC.

To upgrade to the latest firmware version, please visit the AKD BASICsoftware/firmware tab on koll-morgen.com.

To downgrade to an older version of WorkBench, please visit the AKD BASIC archive tab on koll-morgen.com.

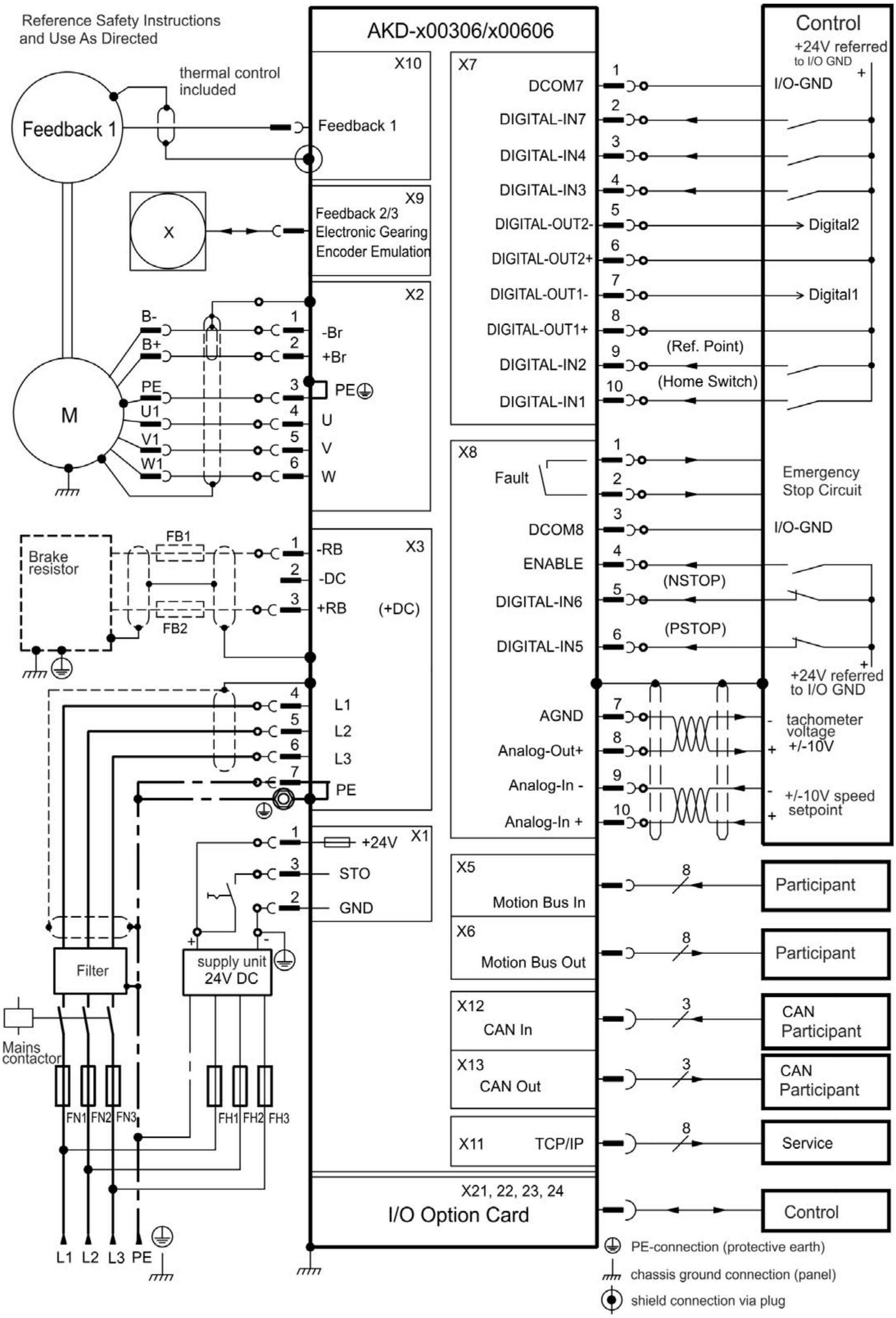
See Also

Unsupported Firmware (pg 282) | Invalid Firmware (pg 282)

22 Connection Diagrams

22.1 Connection Diagram AKD-x00306, AKD-x00606

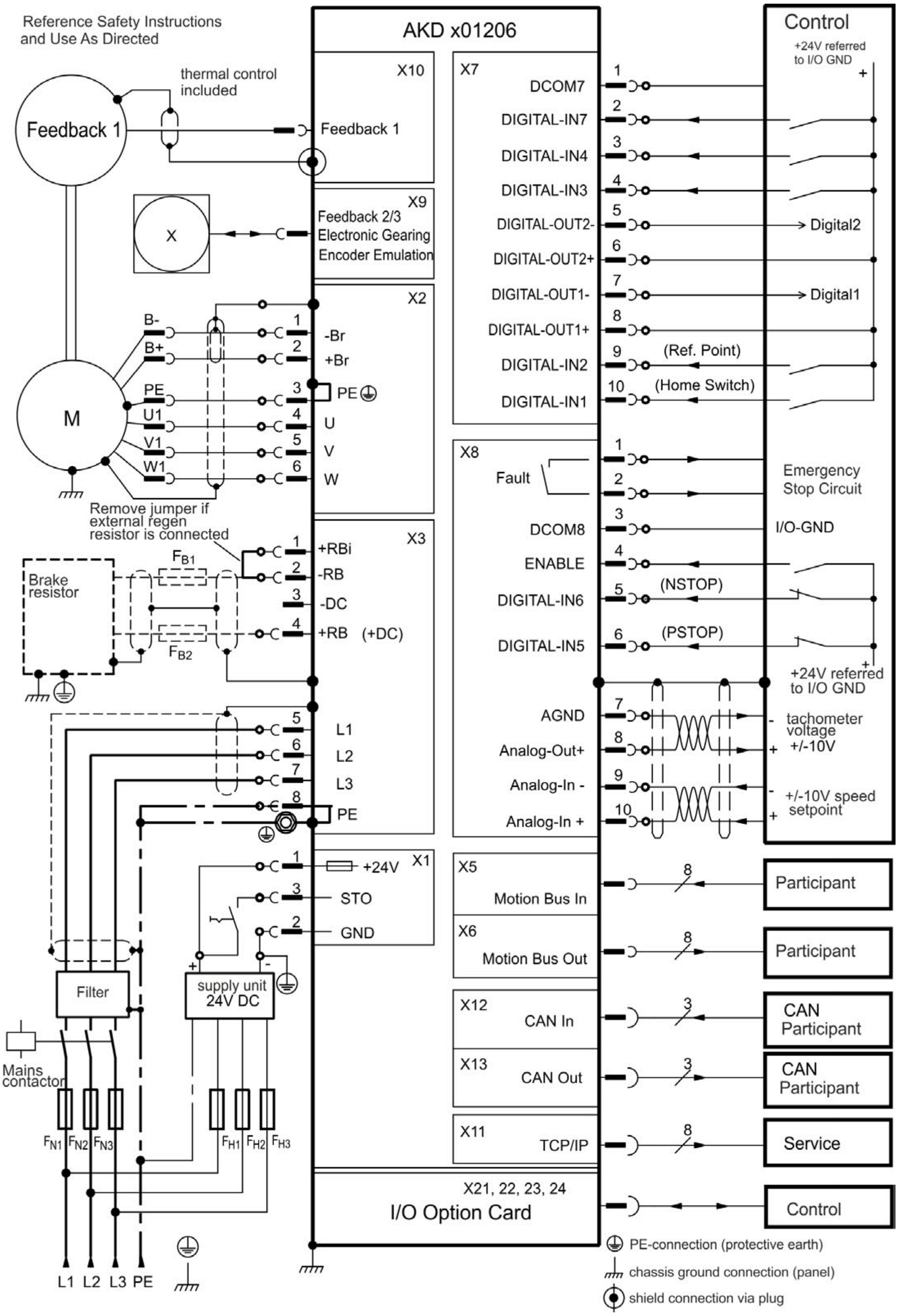
22.2 Wiring Diagram 3 to 6A (230V)



The I/O option is available for AKD-T drives only.

22.3 Connection Diagram AKD-x01206

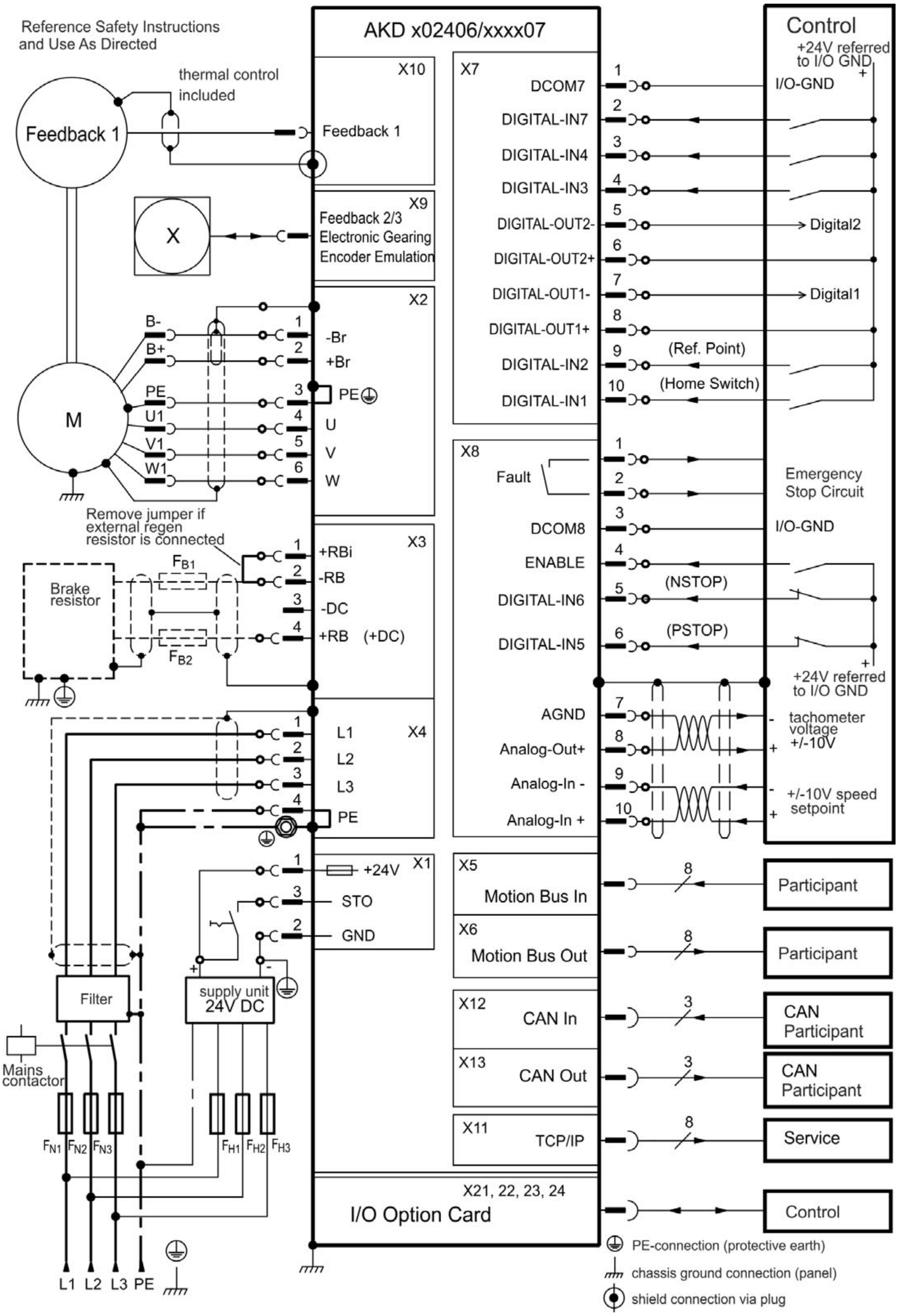
22.4 Wiring Diagram 12A (230V)



The I/O option is available for AKD-T drives only.

22.5 Connection Diagram AKD-x02406 and AKD-xzzz07

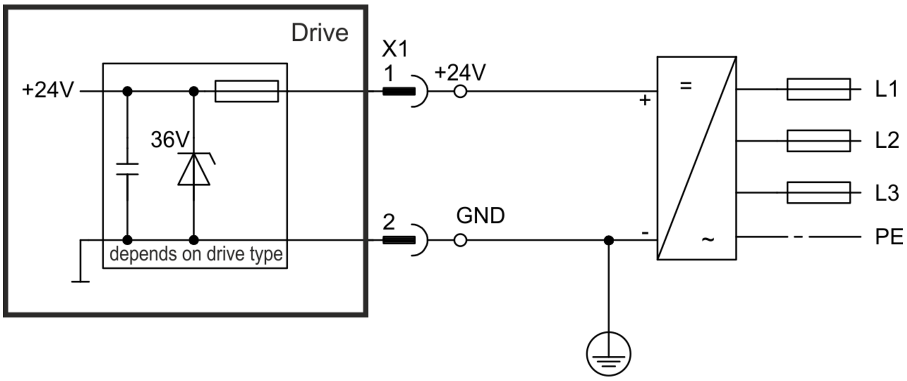
22.6 Wiring Diagram 24A (230V) and 3 to 24 A (480V)



The I/O option is available for AKD-T drives only.

22.7 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).



Pin	Signal	Description
1	+24	+24 Vdc Auxiliary voltage
2	GND	24V Supply GND
3	STO	STO enable (Safe Torque Off)

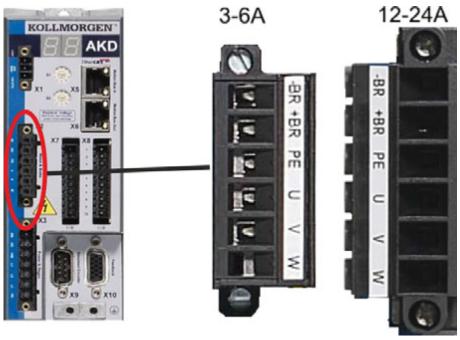
22.8 Motor Connection

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.

NOTICE The dynamic voltage rise can lead to a reduction in the motor operating life and, on unsuitable motors, to flashovers in the motor winding.

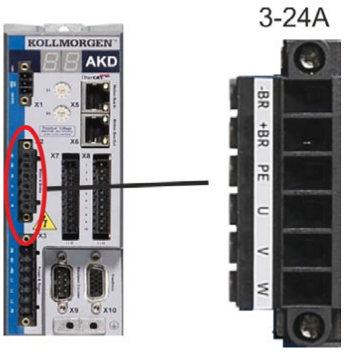
- Only install motors with insulation class F (acc. to IEC60085) or above.
- Only install cables that meet the requirements .

Connector X2 AKD-xzzz06



Pin	Signal	Description
1	-BR	Motor holding brake, negative
2	+BR	Motor holding brake, positive
3	PE	Protective earth (motor housing)
4	U	Motor phase U
5	V	Motor phase V
6	W	Motor phase W

Connector X2 AKD-xzzz07



Pin	Signal	Description
1	-BR	Motor holding brake, negative
2	+BR	Motor holding brake, positive
3	PE	Protective earth (motor housing)
4	U	Motor phase U
5	V	Motor phase V
6	W	Motor phase W

22.9 External Regen Resistor (X3)

For technical data on the brake circuit . Fusing (such as fusible cut-outs) to be provided by the user .

AKD-x00106x00306 to AKD-x00606 (X3)		
Pin	Signal	Description
1	-RB	External Regen Resistor negative
3	+RB	External Regen Resistor positive

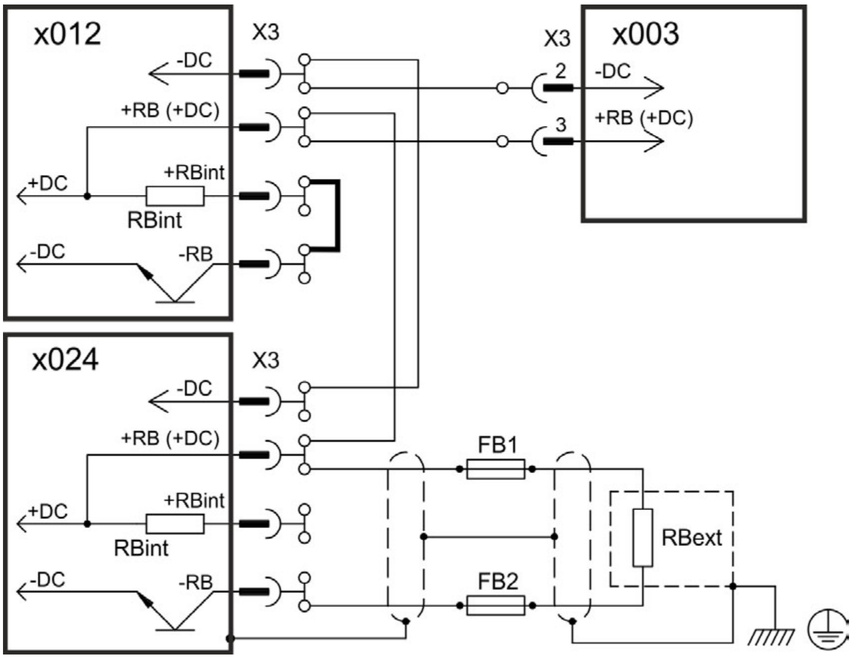
AKD-x01206 (X3)		
Pin	Signal	Description
1	+Rbint	Internal RegenResistor positive
2	-RB	External RegenResistor negative
4	+RB	External RegenResistor positive

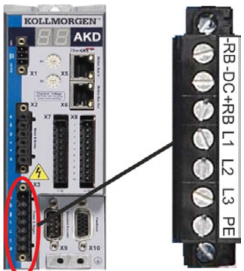
AKD-x02406 & AKD-xzzz07 (X3)		
Pin	Signal	Description
2	-RB	External RegenResistor negative
4	+RB	External RegenResistor positive

22.10 DC Bus Link (X3)

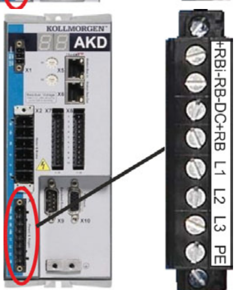
The DC bus link can be connected in parallel so that the regen power is divided between all the drives that are connected to the same DC bus link circuit. Every drive must have it's own power connection to mains voltage, even if the DC bus link is used.

NOTICE The drive can be destroyed if DC bus link voltages are different. Only drives with mains supply from the same mains (identical mains phases and voltage) may be connected by the DC bus link. Use unshielded single cores with a maximum length of 200 mm. Use shielded cables for longer lengths.





AKD-x00106x00306 to AKD-x00606 (X3)		
Pin	Signal	Description
2	-DC	DC-Link Bus negative
3	+DC (+RB)	DC-Link Bus positive



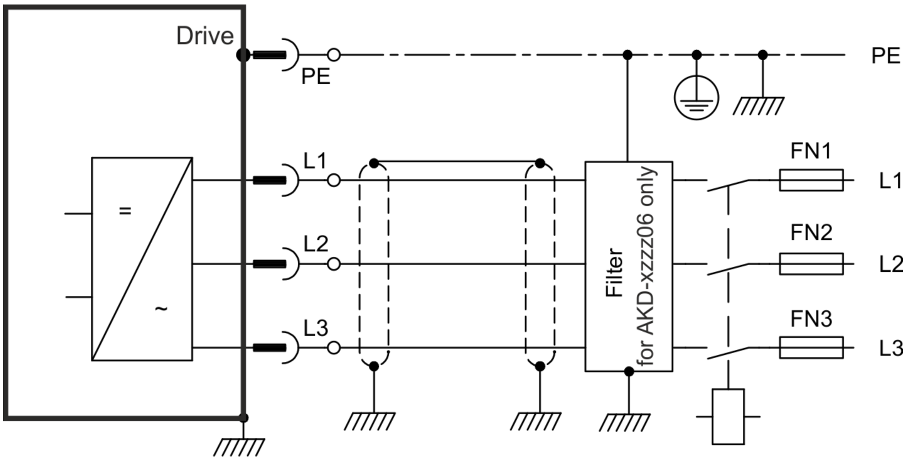
AKD-x01206 (X3)		
Pin	Signal	Description
3	-DC	DC-Link Bus negative
4	+DC (+RB)	DC-Link Bus positive

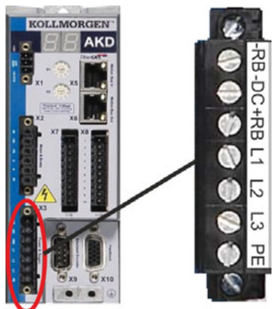
AKD-x02406 & AKD-xzzz07 (X3)		
Pin	Signal	Description
3	-DC	DC-Link Bus negative
4	+DC (+RB)	DC-Link Bus positive

22.11 Mains Supply Connection (X3, X4)

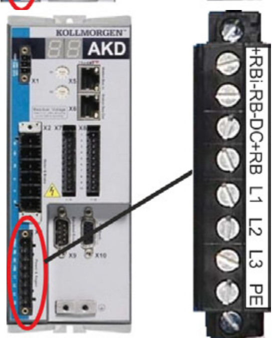
22.11.1 Three Phase connection (all AKD types)

- Directly to 3-phase supply network, supply networks
- Filtering for AKD-xzzz06 to be provided by the user.
- Fusing (such as fusible cut-outs) to be provided by the user .





AKD-x00106x00306 to AKD-x00606 (X3)		
Pin	Signal	Description
4	L1	Line 1
5	L2	Line 2
6	L3	Line 3
7	PE	Protective Earth

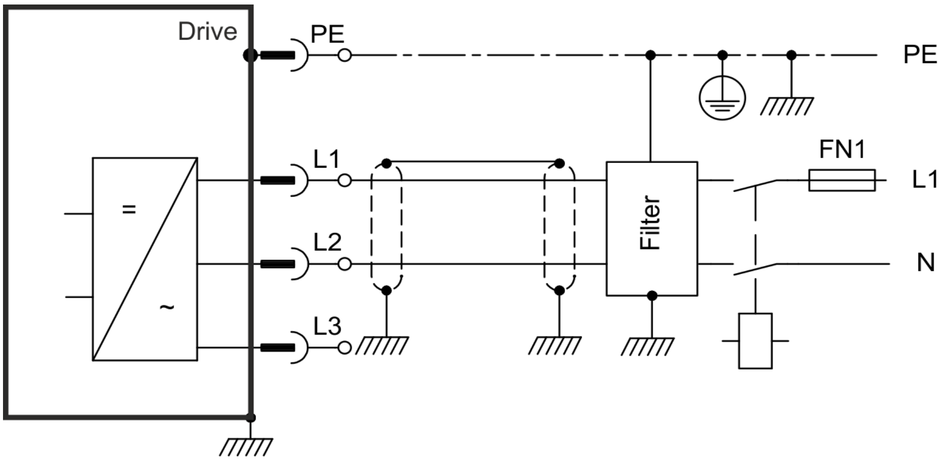


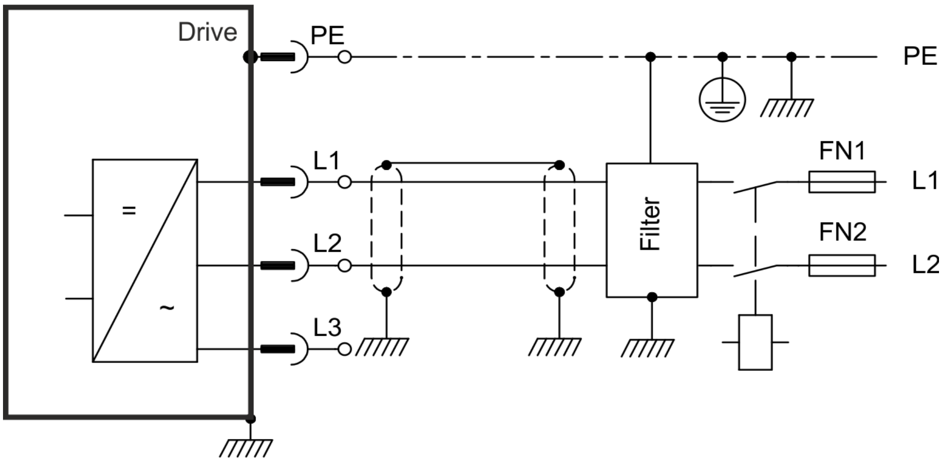
AKD-x01206 (X3)		
Pin	Signal	Description
5	L1	Line 1
6	L2	Line 2
7	L3	Line 3
8	PE	Protective Earth

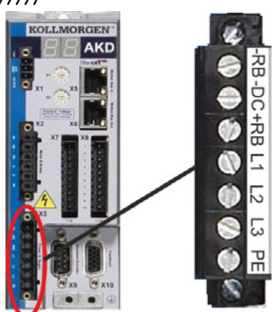
AKD-x02406 & AKD-xzzz07 (X4)		
Pin	Signal	Description
1	L1	Line 1
2	L2	Line 2
3	L3	Line 3
4	PE	Protective Earth

22.11.2 Single phase connection (AKD-x00106x00306 to AKD-x01206 only)

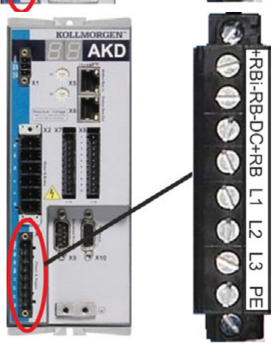
- Directly to single-phase supply network (120 $V_{-10\%}$ to 240 $V^{+10\%}$) with neutral line or Directly to two-phase supply network (120 $V_{-10\%}$ to 240 $V^{+10\%}$) without neutral line
- Supply networks
- Leave L3 open circuit
- Filtering to be provided by the user.
- Fusing (such as fusible cut-outs) to be provided by the user







AKD-x00106x00306 to AKD-x00606 (X3)		
Pin	Signal	Description
4	L1	Line 1
5	L2 (N)	Neutral or Line 2
7	PE	Protective Earth



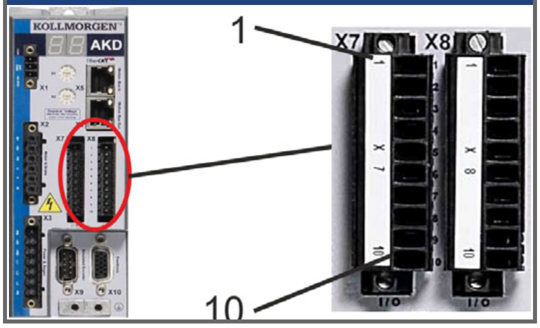
AKD-x01206 (X3)		
Pin	Signal	Description
5	L1	Line 1
6	L2 (N)	Neutral or Line 2
8	PE	Protective Earth

22.12 I/O Connection

22.12.1 I/O Connectors X7 and X8 (all AKD variants)

Standard digital and analog I/O signals are connected to X7 and X8.

AKD-B , -P, -T



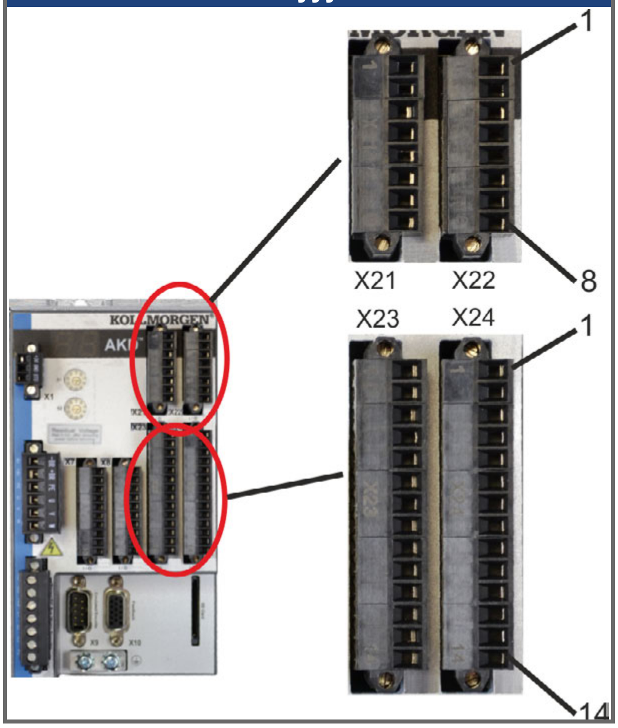
Conn.	Pin	Signal	Abbreviation	Function
X7	1	Digital Common X7	DCOM7	Common line for X7 pins 2, 3, 4, 9, 10
X7	2	Digital Input 7	DIGITAL-IN 7	Programmable
X7	3	Digital Input 4	DIGITAL-IN 4	Programmable
X7	4	Digital Input 3	DIGITAL-IN 3	Programmable
X7	5	Digital Output 2-	DIGITAL-OUT2-	Programmable
X7	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
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.

22.12.2 I/O Connectors X21, X22, X23 and X24 (AKD-T with I/O option card only)

The I/O option card offers four additional connectors X21, X22, X23, X24 for I/O signals.

AKD-xyyyzz-IC



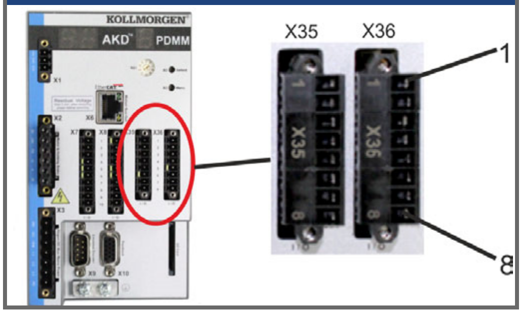
Conn.	Pin	Signal	Abbreviation	Function
X21	1	Digital Input 21	DIGITAL-IN 21	Programmable
X21	2	Digital Input 22	DIGITAL-IN 22	Programmable
X21	3	Digital Input 23	DIGITAL-IN 23	Programmable
X21	4	Digital Common X21/1_3	DCOM21.1_3	Common line for X21 pins 1, 2, 3
X21	5	Digital Input 24	DIGITAL-IN 24	Programmable
X21	6	Digital Input 25	DIGITAL-IN 25	Programmable
X21	7	Digital Input 26	DIGITAL-IN 26	Programmable
X21	8	Digital Common X21/5_7	DCOM21.5_7	Common line for X21 pins 5, 6, 7
X22	1	Digital Input 27	DIGITAL-IN 27	Programmable
X22	2	Digital Input 28	DIGITAL-IN 28	Programmable
X22	3	Digital Input 29	DIGITAL-IN 29	Programmable
X22	4	Digital Common X22/1_3	DCOM22.1_3	Common line for X22 pins 1, 2, 3
X22	5	Digital Input 30	DIGITAL-IN 30	Programmable
X22	6	Digital Input 31	DIGITAL-IN 31	Programmable
X22	7	Digital Input 32	DIGITAL-IN 32	Programmable
X22	8	Digital Common X22/5_7	DCOM22.5_7	Common line for X22 pins 5, 6, 7

Conn.	Pin	Signal	Abbreviation	Function
X23	1	Analog Output 2 +	Analog-Out2	Programmable
X23	2	reserved	n.c.	n.c.
X23	3	Analog Ground	AGND	Programmable
X23	4	reserved	n.c.	n.c.
X23	5	Digital Output 21+	DIGITAL-OUT 21+	Programmable
X23	6	Digital Output 21-	DIGITAL-OUT 21-	Programmable
X23	7	Digital Output 22+	DIGITAL-OUT 22+	Programmable
X23	8	Digital Output 22-	DIGITAL-OUT 22-	Programmable
X23	9	Digital Output 23+	DIGITAL-OUT 23+	Programmable
X23	10	Digital Output 23-	DIGITAL-OUT 23-	Programmable
X23	11	Digital Output 24+	DIGITAL-OUT 24+	Programmable
X23	12	Digital Output 24-	DIGITAL-OUT 24-	Programmable
X23	13	Relay Output 25	DIGITAL-OUT 25	Programmable, relay
X23	14	Relay Output 25	DIGITAL-OUT 25	Programmable, relay
X24	1	Analog Input 2+	Analog-In2+	Programmable
X24	2	Analog Input 2-	Analog-In2-	Programmable
X24	3	Analog Ground	AGND	Programmable
X24	4	reserved	n.c.	n.c.
X24	5	Digital Output 26+	DIGITAL-OUT 26+	Programmable
X24	6	Digital Output 26-	DIGITAL-OUT 26-	Programmable
X24	7	Digital Output 27+	DIGITAL-OUT 27+	Programmable
X24	8	Digital Output 27-	DIGITAL-OUT 27-	Programmable
X24	9	Digital Output 28+	DIGITAL-OUT 28+	Programmable
X24	10	Digital Output 28-	DIGITAL-OUT 28-	Programmable
X24	11	Digital Output 29+	DIGITAL-OUT 29+	Programmable
X24	12	Digital Output 29-	DIGITAL-OUT 29-	Programmable
X24	13	Relay Output 30	DIGITAL-OUT 30	Programmable, relay
X24	14	Relay Output 30	DIGITAL-OUT 30	Programmable, relay

22.12.3 I/O Connectors X35 and X36 (AKD-M only)

AKD PDMM offers two additional connectors X35 and X36 for digital I/O signals.

AKD-M



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.

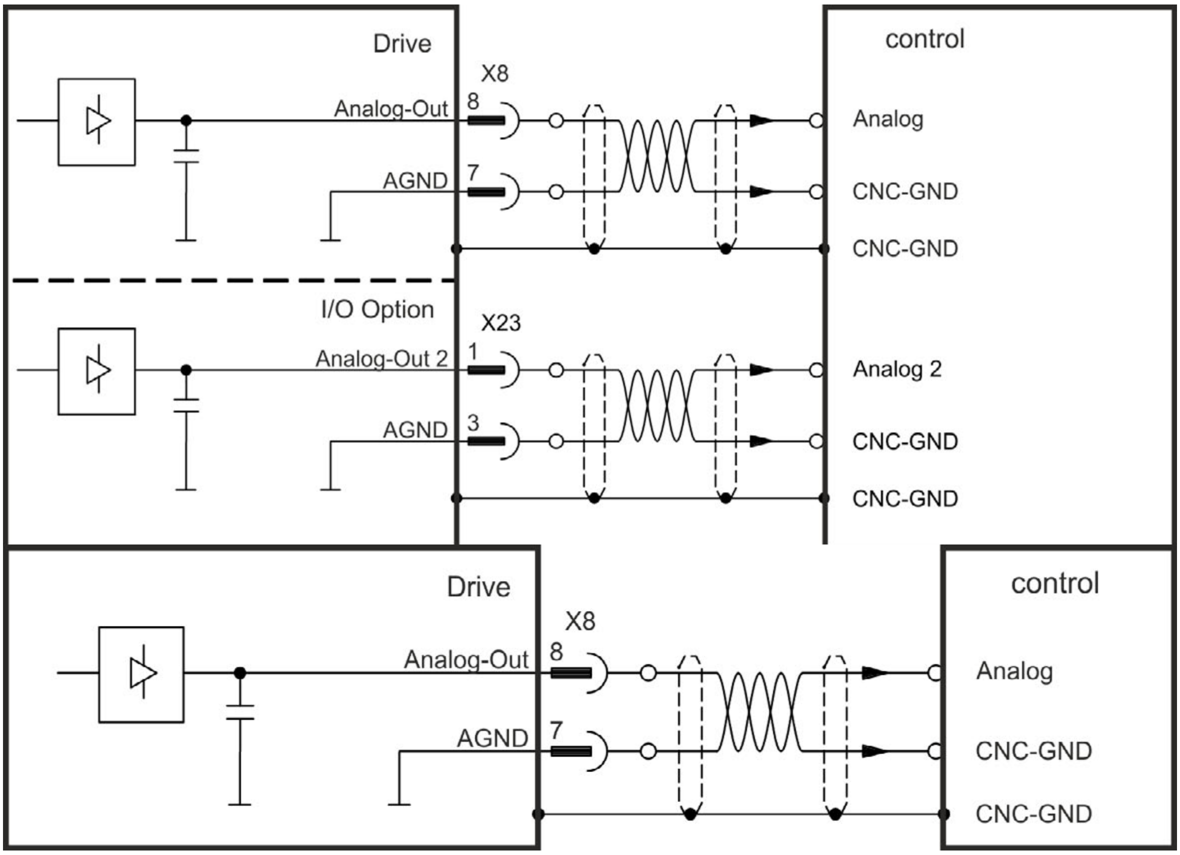
22.13 Analog Output (X8, X23)

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, drives with built-in I/O option card offer a second output on X23. A list of the pre-programmed functions is included in the WorkBench setup software.

Technical characteristics

- Output voltage range referring to AGND: ±10 V
- Resolution: 16 Bit and fully monotonic
- Unadjusted offset: < 50 mV
- Offset drift typ: 250 µV/°C
- Gain or slope tolerance: +/- 3%
- Nonlinearity: < 0.1% of full scale or 10 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



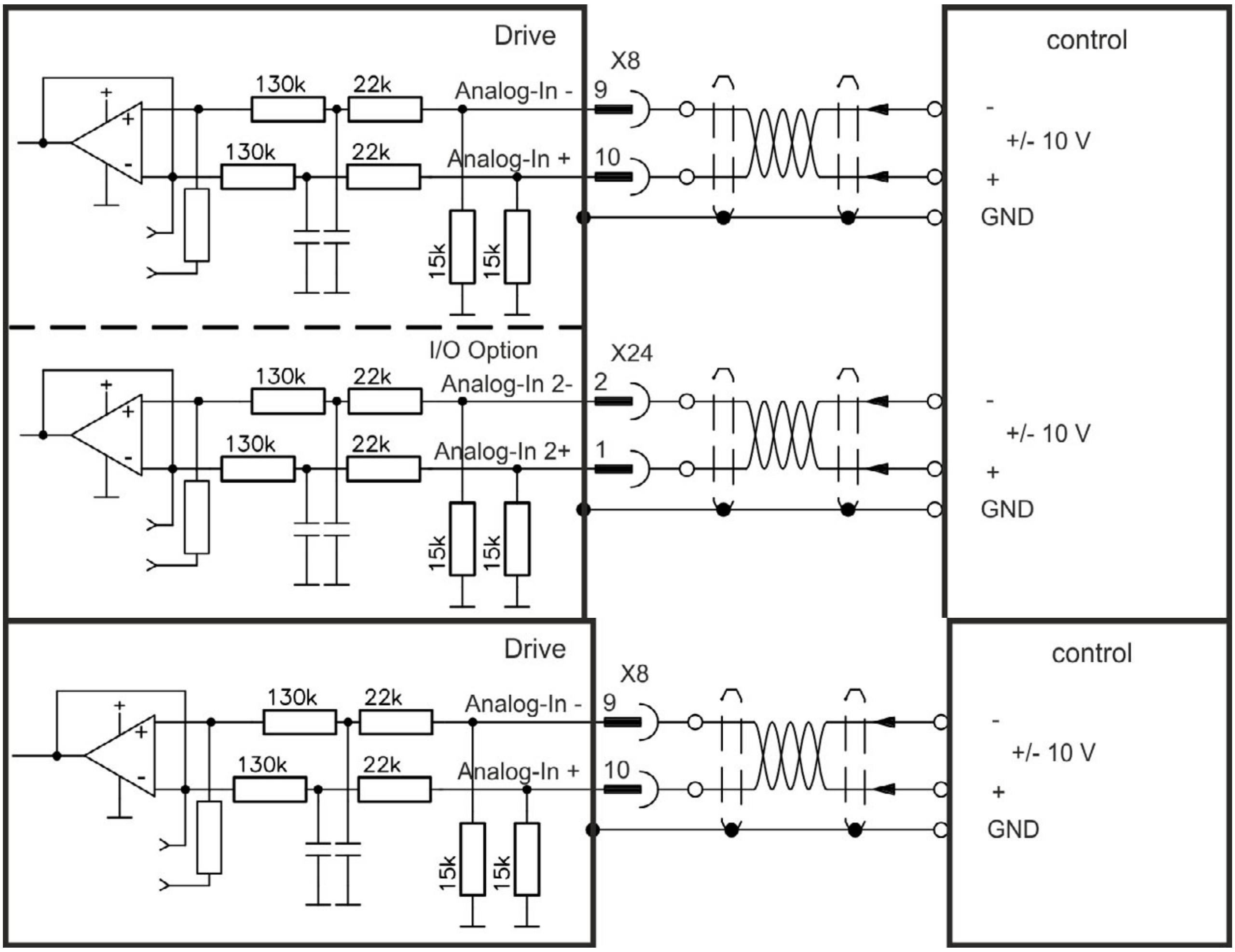
22.14 Analog Input (X8, X24)

The drive is fitted with differential inputs for analog torque, velocity, or position control. The standard drive offers one analog input on X8, drives with built-in I/O option card offer a second input on X24.

Technical characteristics

- Differential input voltage range: ± 12.5 V
- Maximum input voltage referring to I/O Return: -12.5, +16.0 V
- Resolution: 16 Bit and fully monotonic
- Unadjusted offset: < 50 mV
- Offset drift typ: 250 µV / ° C
- Gain or slope tolerance: +/- 3%
- Nonlinearity: < 0.1% of full scale or 12.5 mV
- Common Mode Rejection Ratio: > 30 dB at 60 Hz
- Input impedance: > 13k 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 (-)

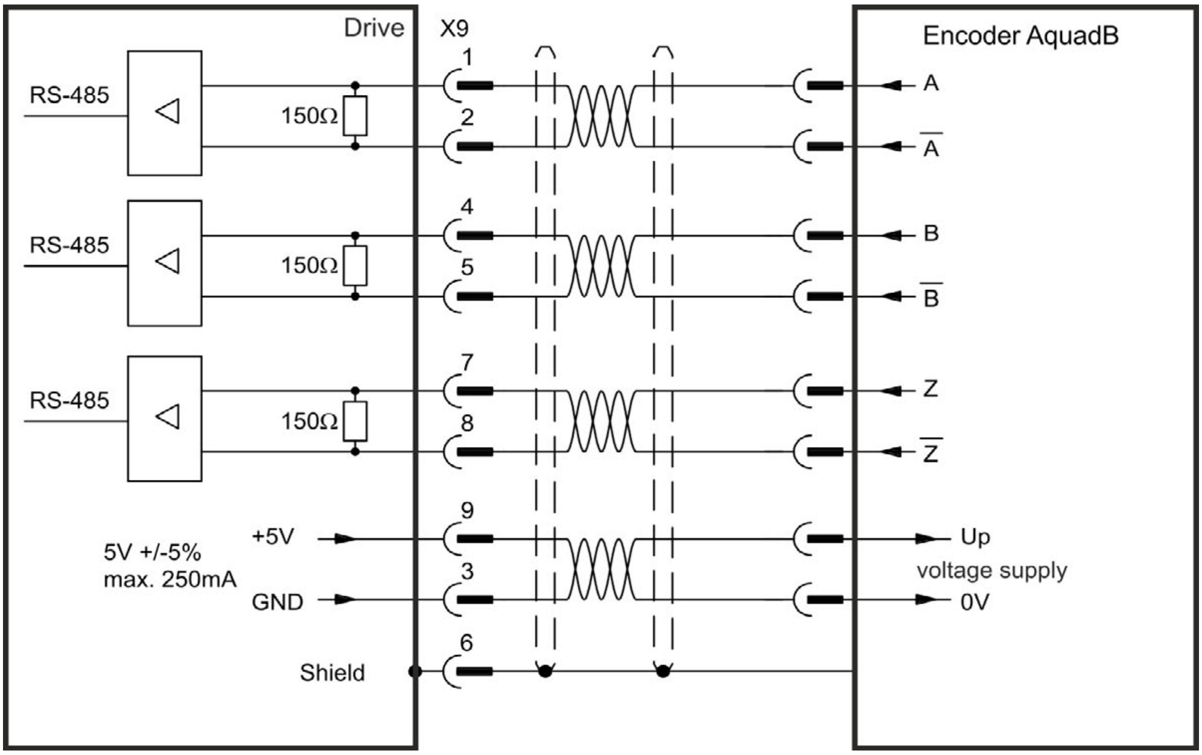
To reverse the direction of rotation, swap the connections to terminals +/-, or change the DRV.DIR parameter in the “Feedback 1” screen page.

22.15 Command encoder signal connection

22.15.1 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. Don't use for primary motor feedback connection!

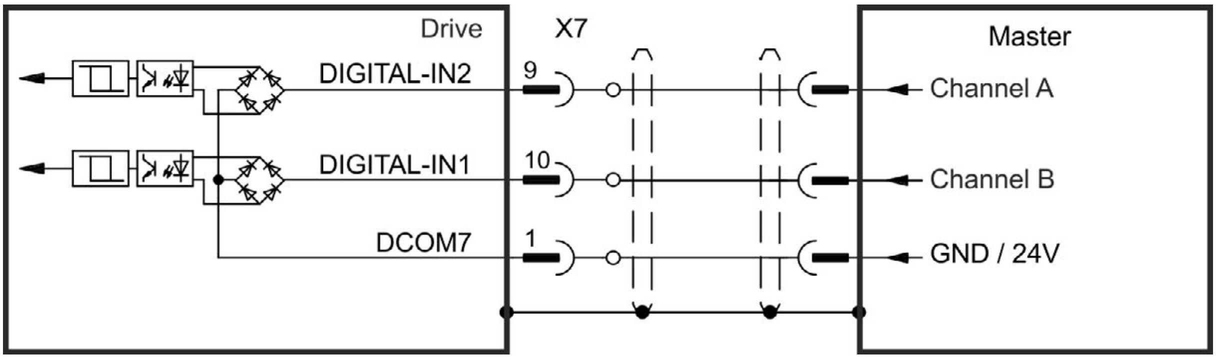
Connection Diagram



22.15.2 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. Don't use for primary motor feedback connection!

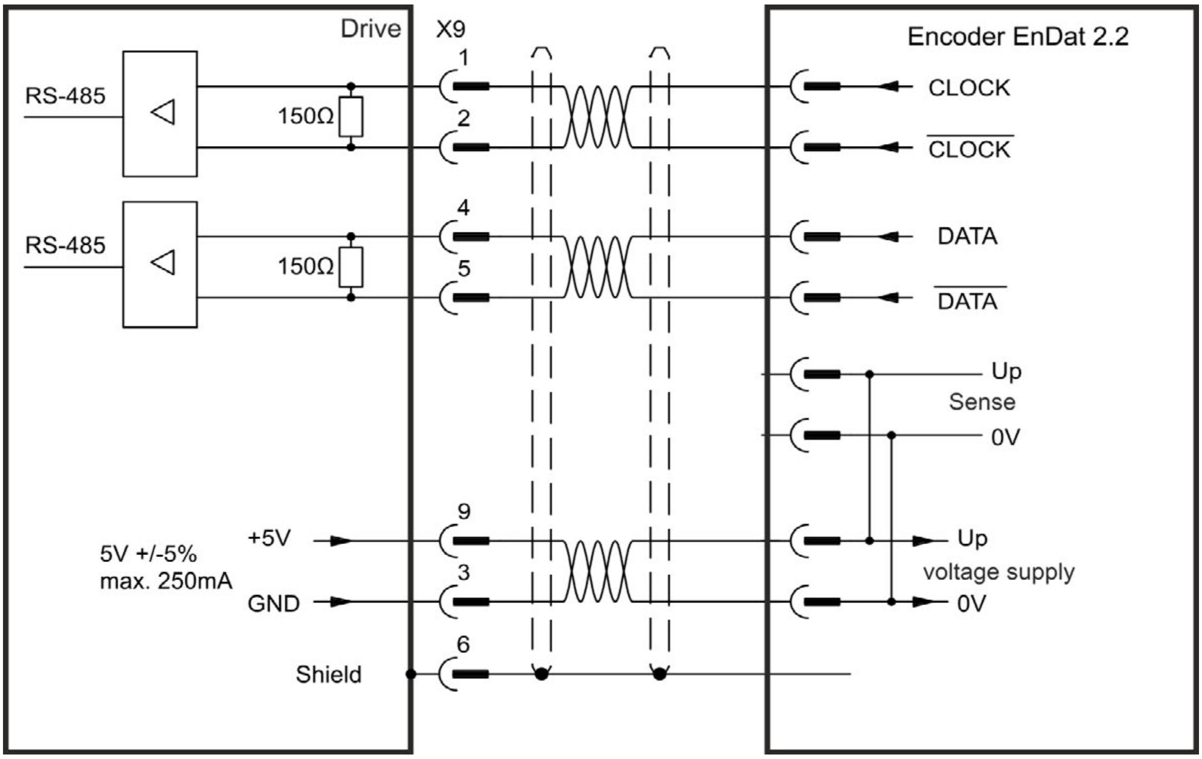
Connection Diagram



22.15.3 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. Don't use for primary motor feedback connection!

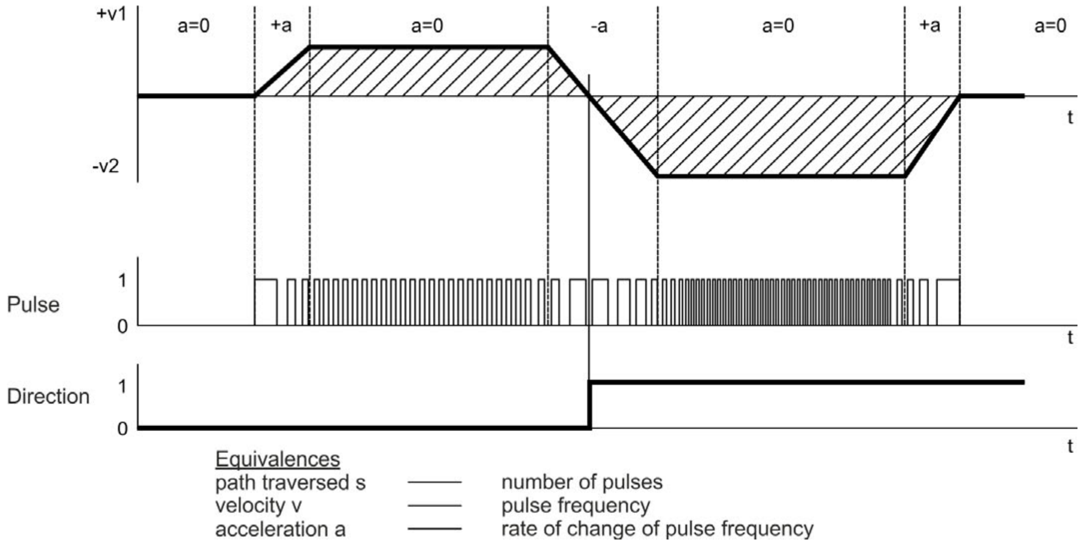
Connection Diagram



22.16 Pulse / Direction signal connection

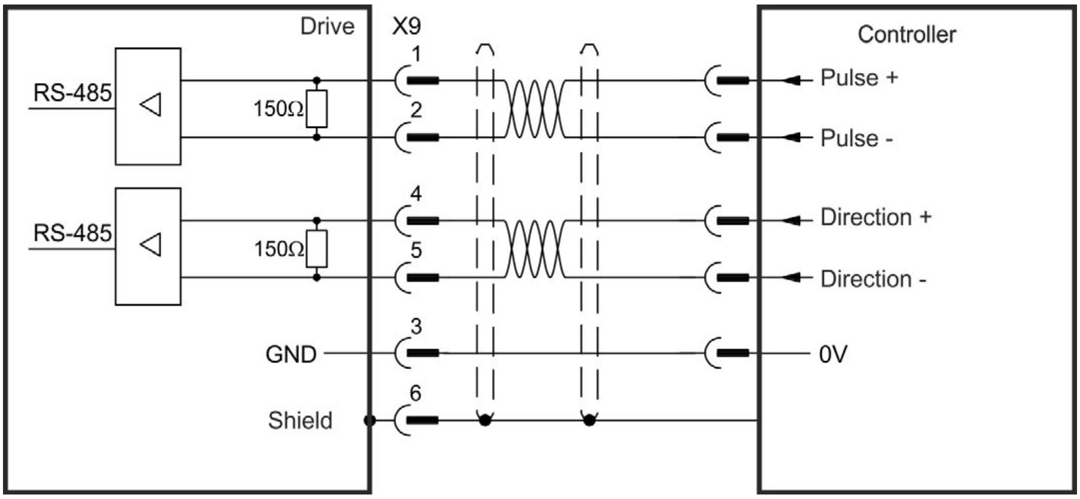
The drive can be connected to a stepper-motor controller. Set parameters for the drive with WorkBench. The number of pulses can be adjusted, so that the drive can be adapted to match any stepper controller.

Speed profile and signal diagram



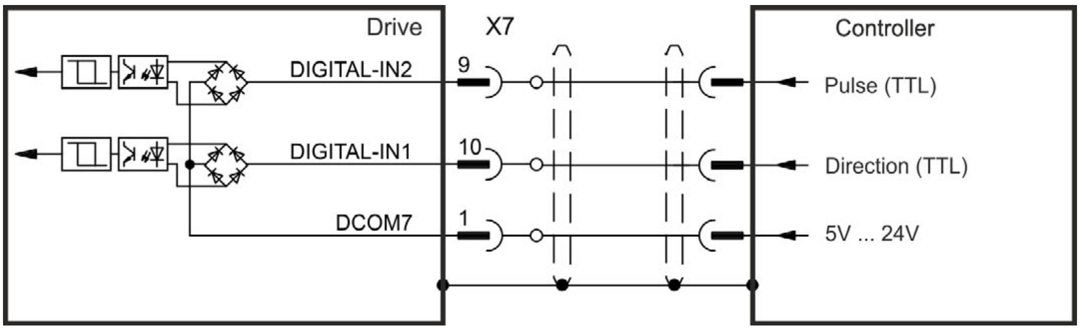
22.16.1 Pulse / Direction input 5 V (X9)

Connection to 5 V signal level stepper-motor controllers.



22.16.2 Pulse / Direction Input 5V (X7)

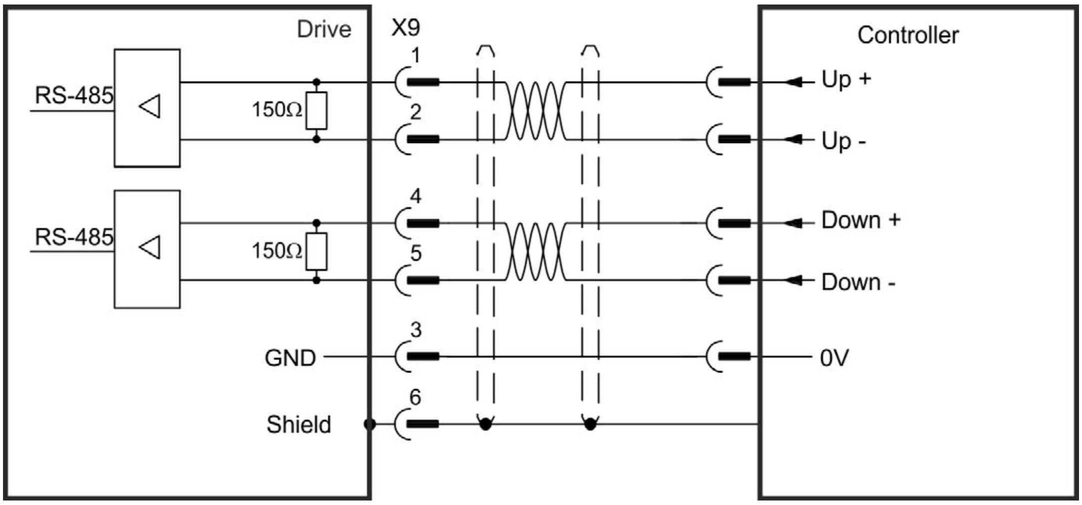
Connection industry standard 5V logic stepper-motor controllers with Pulse/Direction or Step/Direction outputs. Note that the X7 opto inputs can work with 5V up to 24V logic and so these inputs can be driven by 24V logic inputs as well.



22.17 Up / Down signal connection

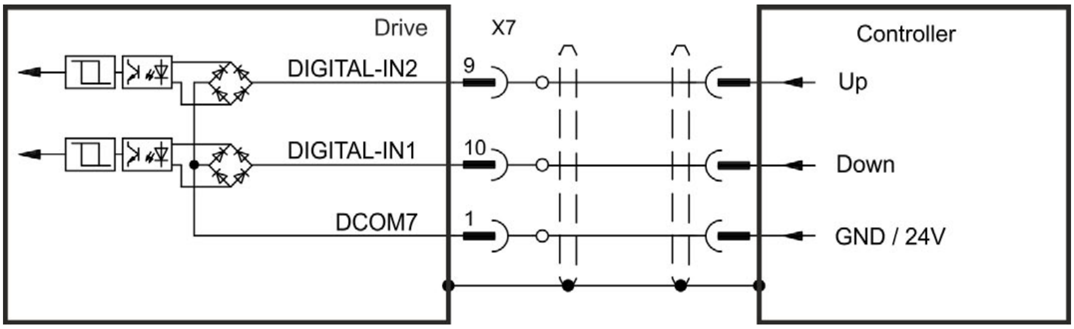
22.17.1 Up / Down input 5 V (X9)

The drive can be connected to a third-party controller which delivers 5 V up-down signals

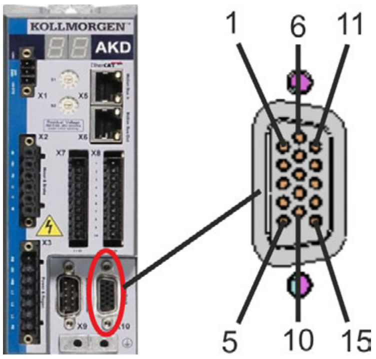


22.17.2 Up / Down input 24 V (X7)

The drive can be connected to a third-party controller which delivers 24 V up-down signals.



22.18 Feedback Connector (X10)



PIN	SFD	DSL	Resolver	BiSS A (analog)	BiSS C (digital)	EnDAT 2.1	EnDAT 2.2	Hiper-face	Sine Enc. +Hall	Tama-gawa Smart Abs	Incr. Enc. +Hall
1	-	-	-	-	-	-	-	-	Hall U	-	Hall U
2	-	-	-	CLK+	CLK+	CLK+	CLK+	-	Hall V	-	Hall V
3	-	-	-	CLK-	CLK-	CLK-	CLK-	-	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+	SD+	Zero+
7	COM-	COM-	R2 Ref-	DAT-	DAT-	DAT-	DAT-	DAT-	Zero-	SD-	Zero-
8	-	-	Thermal control (PTC)								
9	-	-	Thermal control (PTC, GND)								
10	+5 V	+5 V	-	+5 V	+5 V	+5 V	+5 V	+8 to +9 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
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-

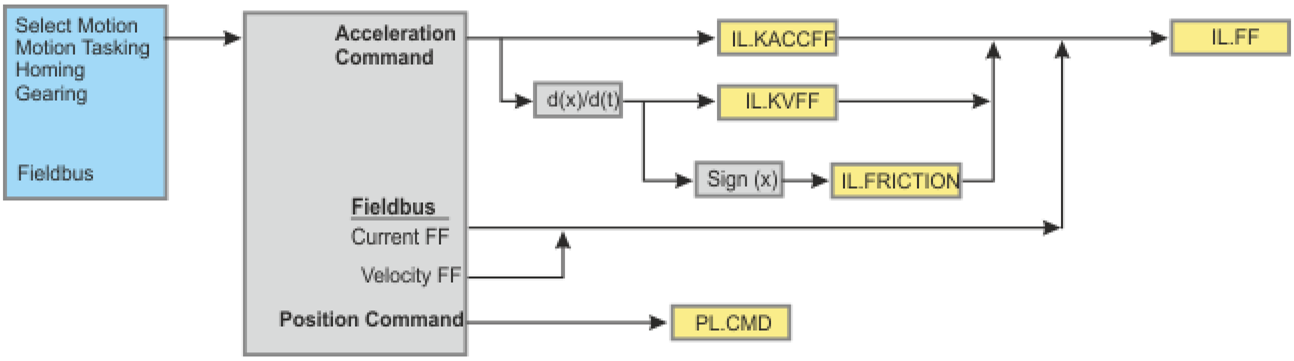
CLK = CLOCK, DAT = DATA, SEN = SENSE

Pin	SFD	DSL	BiSS C (digital)	EnDAT 2.2	Tamagawa Smart Abs	Incremental Encoder +Hall
1	-	-	-	-	-	Hall U
2	-	-	CLOCK+	CLOCK+	-	Hall V
3	-	-	CLOCK-	CLOCK-	-	Hall W
4	SENSE+	-	SENSE+	SENSE+	SENSE+	SENSE+
5	SENSE-	-	SENSE-	SENSE-	SENSE-	SENSE-
6	COM+	COM+	DATA+	DATA+	SD+	Zero+
7	COM-	COM-	DATA-	DATA-	SD-	Zero-
8	-	-	Thermal control (PTC)			
9	-	-	Thermal control (PTC)			
10	+5 V	+5 V	+5 V	+5 V	+5 V	+5 V
11	0 V	0 V	0 V	0 V	0 V	0 V

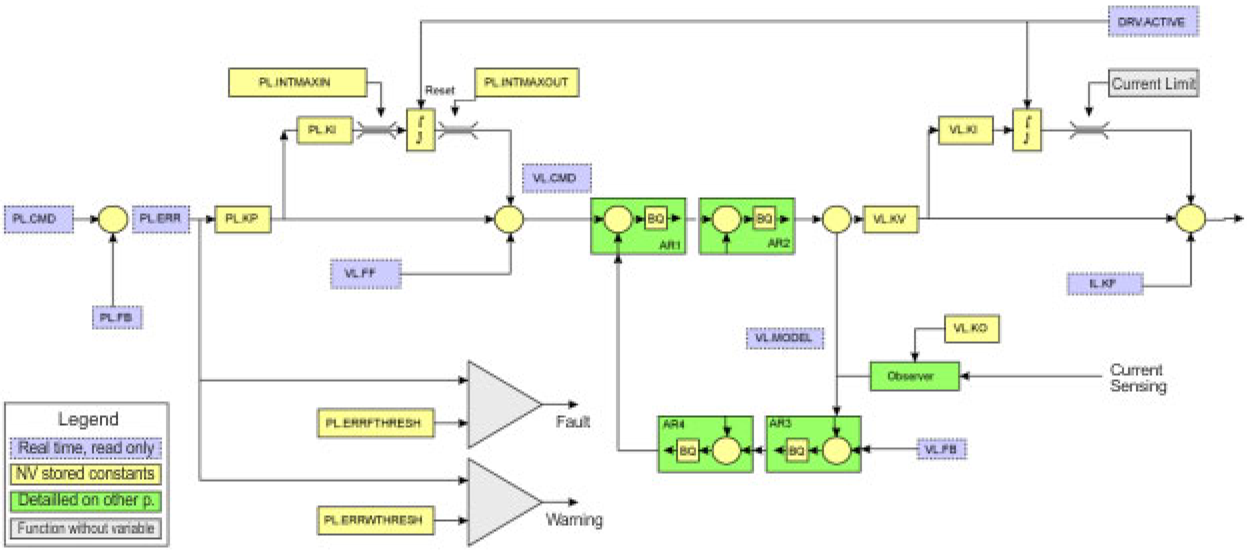
Pin	SFD	DSL	BiSS C (digital)	EnDAT 2.2	Tamagawa Smart Abs	Incremental Encoder +Hall
12	-	-	-	-	-	A+
13	-	-	-	-	-	A-
14	-	-	-	-	-	B+
15	-	-	-	-	-	B-

23 Block Diagrams

23.1 Block Diagram for Current Loop



23.2 Block Diagram for Position/Velocity Loop



24 Fieldbus Manuals

This chapter contains links to the fieldbus manuals for AKD drives.

24.0.1 FieldBus Manuals

The fieldbus manuals describe the different options for connecting and communicating with the AKD drive. Links to these manuals are listed below

24.0.1.1 AKD Modbus Communication

This manual describes Modbus communication, a simple communication protocol used for reporting data from an industrial device to an HMI or PLC. This manual can be found here: Modbus (pg 311)

24.0.1.2 AKD EtherCAT Communication

This manual describes the installation, setup, range of functions, and software protocol for the EtherCAT AKD product series. The manual can be found online at: Kollmorgen.com

A pdf format is also available on the disk included with the drive.

24.0.1.3 AKD CANopen Communication

This manual describes the installation, setup, range of functions, and software protocol for the CANopen AKD product series. The manual can be found online at: Kollmorgen.com

A pdf format is also available on the disk included with the drive.

24.0.1.4 AKD PROFINET

This manual describes the installation, setup, range of functions, and software protocol for the PROFINET AKD product series. The manual can be found online at: Kollmorgen.com

A pdf format is also available on the disk included with the drive.

24.0.1.5 AKD SynqNet Communication

This manual describes the installation, setup, range of functions, and software protocol for the SynqNet AKD product series. The manual can be found online at: Kollmorgen.com

A pdf format is also available on the disk included with the drive.

24.0.1.6 AKD EtherNet/IP Communication

This manual describes the installation, setup, range of functions, and software protocol for the EtherNet/IP AKD product series. The manual can be found online at: Kollmorgen.com

A pdf format is also available on the disk included with the drive.

24.0.1.7 Modbus

Overview

Modbus is a simple communication protocol often used for reporting data from an industrial device to an HMI (see HMI Modbus Communication with AKD)or PLC. Modbus TCP extends the protocol to TCP/IP networks by embedding the same Protocol Data Unit within TCP/IP packets. The AKD supports a Modbus TCP service channel for connections with up to three masters.

Most drive parameters are supported over Modbus TCP (see Modbus Parameter Table) with the exception of commands which output character strings. For information about the Modbus protocol, please see: http://www.Modbus.org/specs.php.

Modbus Installation and Setup

Modbus TCP is provided over the service port on the top of the drive (X11 connector, the connector used for WorkBench). Connect the drive and a device such as an HMI to a working Ethernet network. For ease of testing and configuration, connect a PC running WorkBench to the same network.

After booting, the drive will flash its Ethernet IP address on the front display. The drive can be accessed at this address for Modbus on port 502. WorkBench uses the same address, but a different port number.

Once the devices are connected, the connected device can open a connection to the AKD using these settings:

- IP Address: read from drive display or Workbench connect screen
- Port: 502
- Add Modbus CRC code: No

Overview of Messaging

All parameters with an internal data width of 32-bit and lower are mapped to Modbus as 32-bit (2 register) values. The contents of both registers must be read or written in the same message, by setting the Starting Address to the lowest index of the parameter and the Quantity of Registers to two.

All parameters with an internal data width of 64-bit are mapped to Modbus as 64-bit (4 register) values. The contents of all four registers must be read or written in the same message by setting the Starting Address to the lowest index of the parameter and the Quantity of Registers to four.

Parameters are scaled as in the Terminal window of WorkBench, but note that floating point values are scaled by an additional factor of 1,000 to retain precision.

NOTE A read attempt of command or write-only parameters will return a value of zero.

Supported Functions

Two Modbus functions are currently supported:

- Read Holding Registers. Function code = 0x03 (3).
- Write Multiple Registers. Function code = 0x10 (16).

To allow only 32-bit and 64-bit access, the quantity of registers is limited to two registers for 32-bit variables and four registers for 64-bit variables. You can also read/write to a single register for certain parameters. Other quantities return an exception response.

Read Holding Registers (0x03)

This function code is used to read all registers of one drive parameter.

Request

Function Code	1 Byte	0x03

Starting Address	2 Bytes	0x0000-0xFFFF (see Modbus Parameter Table)
Quantity of Registers	2 Bytes	Two 32-bit values or four 64-bit values

Normal Response

Function Code	1 Byte	0x03
Byte Count	1 Byte	2 x N*
Register Value	N* x 2 Bytes	Data

*N = Quantity of Registers

Error Response

Function Code	1 Byte	0x83
Exception Code	1 Byte	See Exception Response Codes (pg 313)

Following is an example of reading the position loop feedback PL.FB at index 588 (0x024C), with an actual value of 0x1 2A05 F200.

Request

Function	03
Starting Address	02 4C
Quantity of Registers	00 04

Response

Function	03
Byte Count	08
Register 588	00 00
Register 589	00 01
Register 590	2A 05
Register 591	F2 00

Write Multiple Registers (0x10)

This function code is used to write all registers of one drive parameter.

Request

Function Code	1 Byte	0x10
Starting Address	2 Bytes	0x0000-0xFFFF (see Modbus index in parameter lists)
Quantity of Registers	2 Bytes	2 (32-bit values) or 4 (64-bit values)
Byte Count	1 Byte	2 x N*
Register Value	N* x 2 Bytes	Data

*N = Quantity of Registers

Normal Response

Function Code	1 Byte	0x10
Starting Address	2 Bytes	0x0000-0xFFFF
Quantity of Registers	2 Bytes	2 or 4

Error Response

Function Code	1 Byte	0x90
Exception Code	1 Byte	See Exception Response Codes (pg 313)

Following is an example of setting the mode of digital input 1 (DIN1.MODE) at index 122 (0x007A) to a value of 3.

Request

Function	10
Starting Address	00 7A
Quantity of Registers	00 02
Byte Count	04
Register 122	00 00
Register 123	00 03

Response

Function	10
Starting Address	00 7A
Quantity of Registers	00 02

Exception Response Codes

Standard Codes

Description	Exception Code
Illegal Function	1
Illegal data address	2
Illegal Data Value	3
Slave Device Failure	4
Acknowledge	5
Slave Device Busy	6
Memory Parity Error	8
Gateway Path Unavailable	10
Gateway Target Device Failed To Respond	11

Manufacturer Codes

Description	Exception Code
Illegal Block Size (must be two registers for 32-bit and four registers for 64-bit parameters)	32
Unknown Parameter (bad address)	33
Error Processing Command (such as "Data value is greater than maximum")	34
Byte Count field is invalid in request	35

Reading Product Serial Numbers Over Modbus

The following register addresses are used to read a serial number over Modbus.

Description	Modbus Register Address	Attribute
Amount of registers	6000	16 bit
Characters 1 and 2	6001	16 bit
Characters 3 and 4	6002	16 bit
Characters 5 and 6	6003	16 bit
Characters 7 and 8	6004	16 bit
Characters 9 and 10	6005	16 bit
Characters 11 and 12	6006	16 bit

For example, the product serial number "R-6789-12345" can be read as follows.

Modbus Register Address	Value
6000	6
6001	0x522D ("R-")
6002	0x3637 ("67")
6003	0x3839 ("89")
6004	0x2D31 ("-1")
6005	0x3233 ("23")
6006	0x3435 ("45")

Modbus Dynamic Mapping

Modbus dynamic mapping allows you to map any of the fixed register addresses to a new register address. By re-ordering the sequence block, read/write access of the re-mapped parameters becomes possible.

In general, all parameters are mapped as 32-bit values and occupy at least two Modbus registers. Parameters with a 64-bit data width occupy four Modbus registers. All 64-bit parameters are also mapped as 32-bit value (two registers) starting at register address 2000 (see Modbus 64-bit Parameters to 32-bit Mapping (pg 337)).

Configuring Dynamic Mapping

The start address for dynamically mapped registers is 8192 (0x2000).

Mapping works as follows:

1. Enable dynamic mapping by writing a 1 (as a 32-bit value) to register address 4096 (0x1000). Writing a 1 enables the configuration mode of dynamic mapping. Writing a zero puts it into "run" mode, in which you can use the mapped parameter.
2. The mapping is now register oriented. Next, write the valid fixed register address of the parameter you would like to map to an address starting at 8192. To map a 32-bit parameter, you must map the corresponding two registers.

   Example

   AOUT.VSCALE

   Write 36 (hi-word) to address 8192 and 37 (lo-word) to address 8193.

   The register oriented mapping allows you to optimize the data width for block transfers. To map a parameter with a data width of 16-bit, only the lo-word register needs to be mapped.
3. **Example**

   MODBUS.DIO

   Write 941 to address 8194

   MODBUS.DIO fixed addresses are 940 (hi-word) and 941 (lo-word). Only the lo-word is of interest (16-bit value), because only the lo-word contains the actual value.
4. Disable dynamic mapping by writing a 0 (as a 32-bit value) to register address 4096 (0x1000).

Disabling dynamic mapping puts it in a mode where you can now access the value of the mapped parameter and use the mapped register.

The fixed mapping starting at address 0 is still valid.

An example of the entire mapping process is included below:

Example

To re-map MODBUS.DIO (16-bit), MODBUS.DRVSTAT (16-bit), and DRV.MOTIONSTAT (32-bit):

1. Enable dynamic mapping.
2. Block write transfer to address 0x2000 (8192).

   Values: 941, 945, 268, 269
3. Modbus telegram:

   Function: 0x10; Address: 0x20, 0x00; Number of Registers: 0x00, 0x04; Data Length in Byte: 0x08; Data: 0x03, 0xAD, 0x03, 0xB1, 0x01, 0x0C, 0x01, 0x0d
4. Disable dynamic mapping.
5. Execute DRV.NVSAVE to store the mapping table non-volatile.
6. The re-mapped addresses are:

   0x2000 : MODBUS.DIO

   0x2001 : MODBUS.DRVSTAT

   0x2002 : DRV.MOTIONSTAT (hi-word)

   0x2003 : DRV.MOTIONSTAT (lo-word)

Saving and Resetting Dynamic Mapping

Use the following parameters to save and reset dynamic mapping:

- DRV.NVSAVE. Save parameters AND the dynamic mapping table non-volatile.
- DRV.RSTVAR. Set parameters to default value AND clear the dynamic mapping table.

Modbus Dynamic mapping through WorkBench Terminal

WorkBench provides an interface for configuring dynamic mapping. This allows you to save the dynamic mapping table in the WorkBench parameter file.

24.0.2 Modbus Overview

A value is identified through a register address and each register is a 16 bit word. To guarantee backwards compatability, each AKD parameter must have a fixed register address. However, the amount of registers needed to access a parameter depends on the parameter's data type. So if a parameter's data type is changed, the number of registers may change as well, which will result in different register addresses for all the following parameters. To avoid these register shifts, all parameters are mapped as either 32 bit words (2 registers) or 64 bit words (4 registers).

Modbus data is specified as big endian. A lower register address holds a higher rank of significance.

32 bit example:

Register address 0 → high 16 bit word

Register address 1 → low 16 bit word

24.0.3 Dynamic Mapping via Telnet

The following parameters can be set for dynamic mapping in WorkBench.

Parameter	Function
MODBUS.DYNMAP1	Enable dynamic mapping.
MODBUS.DYNMAP 0	Disable dynamic mapping.
MODBUS.CLRDYNMAP	Clear dynamic mapping table.

Map a complete parameter using register address. Map all registers from high to low.

Example:

PL.CMD (register address 570)

```
MODBUS.DYNMAP 1
MODBUS.CLRDYNMAP (optional)
MODBUS.ADDR8192 570
MODBUS.ADDR8193 571
MODBUS.ADDR8194 572
MODBUS.ADDR8195 573
...
MODBUS.DYNMAP 0
```

Map only the significant register from an 8 bit parameter. CAP0.EVENT, register address 58, is mapped as a 32 bit value. 58 is the high 16 bit word, and 59 is the low 16 bit word containing the parameter value. Only register 59 needs to be mapped:

```
MODBUS.ADDR8196 59
```

Scaling Parameters

You can scale parameters accessed through Modbus. This scaling is independent from the units of UNIT.PIN and UNIT.POUT. Instead, the user units are defined by MODBUS parameters:

- MODBUS.PIN (default = 1)
- MODBUS.POUT (default = 1)
- MODBUS.PSCALE (default = 20)

With the default settings shown above, the units are as follows:

- Position [counts (PSCALE bit per rev)]
- Velocity [counts/sec]
- Acceleration / Deceleration [counts/sec2]
- Current (Torque) [mA]

24.0.4 Modbus scaling example

If MODBUS.PSCALE = 20 then all position information is 20 bits.

```
POSITION = 2^20 COUNTS/REV
VELOCITY = 2^20 COUNTS/SEC
ACC/DEC = 2^20/SEC^2
```

This means that there will be 2^20 or 1,048,576 position units per rev. This affects all parameters (velocity is in position units/sec, accel and decel are in position units/sec^2)

NOTE With 20 bits per rev, and 32 bits of data available, you can have 32 bits-20 bits = 12 bits of revs (4096 revs) before the 32 bit data is full and goes negative.

The other scaling factors in Modbus are MODBUS.PIN and MODBUS.POUT.

MODBUS.PIN and MODBUS.POUT are the scaling ratio to help relate the Modbus position resolution to the drive position units resolution (or simply scale Modbus User Units). It allows you to work in user units through Modbus.

For example, if you set up the drive user units to be 10,000 counts/rev:

```
UNIT.PIN = 10,000
UNIT.POUT = 1
```

Then to have Modbus reflect the same units, set up the Modbus units:

```
MODBUS.PIN = 2^MODBUS.PSCALE = 1,048,576
MODBUS.POUT = UNIT.PIN/UNIT.POUT * 1,000 = 10,000,000
```

```
'use this also to account for the lack of a decimal point in Modbus
data
```

Then in Modbus:

```
Counts/rev = 10,000
' In Modbus you would read 10,000,000 but the data represents 10,
000.000 since Modbus shows only integers.
```

Modbus specific registers (Parameters)

The following parameters provide a shortcut to digital I/O, status information, and commands.

Parameter	Bit
MODBUS.DIO	Bit 0 to 6: DIN.STATES Bit 16 and 17: DOUT.STATES
MODBUS.DRVSTAT	Parameters collected: Bit 0: DRV.ACTIVE (drive active) Bit 1: STO.STATUS (STO status) Bit 2: HWLS.POSSTATE (positive HW limit) Bit 3: HWLS.NEGSTATE (negative HW limit) Bit 4: SWLS.STATE (positive SW limit) Bit 5: SWLS.STATE (negative SW limit)
MODBUS.DRV	Bit 0: DRV.STOP (write 1 to execute) Bit 1: DRV.EN (write 1 to enable drive) and DRV.DIS (write 0 to disable drive)
MODBUS.HOME	Bit 0: HOME.MOVE (write 1 to execute) Bit 1: HOME.SET (write 1 to execute)
MODBUS.MOTOR	Bit 0: MOTOR.BRAKE Bit 1: MOTOR.BRAKERLS
MODBUS.MT	Bit 0: MT.CLEAR number from MT.NUM Bit 1: MT.CONTINUE Bit 2: MT.LOAD number from MT.NUM Bit 3: MT.SET Bit 4: MT.MOVE (number from MT.NUM) Note: when both bit 2 and 4 are set then MT.SET is executed first and MT.MOVE second to start the motion task.
MODBUS.SM	Bit 0: One direction (sets SM.MODE to either 0 or 1) Bit 1: Start move Edge-triggered: • 0 → 1 : Start motion (execution of SM.MOVE) • 1 → 0 : Stop motion (execution of DRV.STOP)

32-bit versus 16-bit Values

Modbus mirrors all parameters either as 32-bit or 64-bit values. The minimum data width of 32-bit supports backward compatibility, which means that if the internal data width changes, this change has no effect on the register address of consecutive parameters. Dynamic mapping allows you to map only the registers that are relevant to the application and thus reduces communication overhead.

Mapping of 64-bit Parameters to 32-bit Parameters

All 64-bit parameters are mapped as 32-bit parameters starting at register address 2000. Only the 32-bit lo-word is mapped with the same scaling as the 64-bit parameter. Chapter 7 contains the general mapping table (chapter 7.2) and the 64-bit to 32-bit mapping table (chapter 7.3).

Fault Registers

Fault registers MODBUS.FAULT1 (register address 954) to MODBUS.FAULT10 (register address 972) contain the fault state of the drive.

You can obtain the current fault state as follows:

1. Read MODBUS.FAULT1.
2. If MODBUS.FAULT1 is zero, then the drive is fault free.
3. A nonzero value of MODBUS.FAULT1 is a fault number.
4. If MODBUS.FAULT1 is nonzero, the following fault registers (MODBUS.FAULT2 to MODBUS.FAULT10) contain possible further fault numbers.
5. A value of zero indicates no further faults

Mapping Table

The tables below show Modbus addresses and attributes. The attributes are described as follows:

Attribute	Description
64-bit Parameter	Parameter is internally a 64-bit parameter.
8-bit, 16-bit, 32-bit	Internal data size, mapped as 32-bit (2 registers).
64-bit	Internal data size, mapped as 64-bit (4 registers).
Low 32 bit word	Internally a 64-bit value, only the low 32-bit word is mapped (2 registers).
Signed	Sign bit is significant (negative/positive values are accepted).
Command	Executes a command.
Command, data width	Executes a command with numerical argument (for example, MT.NUM).

Modbus Parameter Table

Parameter	Modbus Register Address	Is 64-bit?	Attributes
AIN.CUTOFF (pg 365)	0		32-bit
AIN.DEADBAND (pg 366)	2		16-bit
AIN.ISCALE (pg 370)	4		32-bit
AIN.OFFSET (pg 372)	6		16-bit, signed
AIN.PSCALE (pg 373)	8	Yes	64-bit, signed
AIN.VALUE (pg 375)	12		16-bit
AIN.VSCALE (pg 376)	14		32-bit
AIN.ZERO (pg 378)	16		Command
AOUT.ISCALE (pg 395)	18		32-bit
AOUT.MODE (pg 396)	20		16-bit
AOUT.OFFSET (pg 398)	22		16-bit, signed
AOUT.PSCALE (pg 399)	24	Yes	64-bit
AOUT.VALUE (pg 401)	28	Yes	64-bit, signed
AOUT.VALUEU (pg 402)	32	Yes	64-bit, signed
AOUT.VSCALE (pg 403)	36		32-bit
BODE.EXCITEGAP (pg 412)	38		8-bit
BODE.FREQ (pg 413)	40		32-bit
BODE.IAMP (pg 414)	42		32-bit, signed
BODE.INJECTPOINT (pg 417)	44		8-bit
BODE.MODE (pg 418)	46		8-bit
BODE.MODETIMER (pg 422)	48		32-bit
BODE.PRBDEPTH (pg 424)	50		8-bit
BODE.VAMP (pg 425)	52	Yes	low 32-bit, signed
CAP0.EDGE (CAP0.EDGE, CAP1.EDGE (pg 431))	54		8-bit
CAP0.EN (CAP0.EN, CAP1.EN (pg 432))	56		8-bit
CAP0.EVENT (CAP0.EVENT, CAP1.EVENT (pg 433))	58		8-bit
CAP0.FILTER (CAP0.FILTER, CAP1.FILTER (pg 436))	60		8-bit
CAP0.MODE (CAP0.MODE, CAP1.MODE (pg 437))	62		8-bit
CAP0.PLFB (CAP0.PLFB, CAP1.PLFB (pg 438))	64	Yes	64-bit, signed
CAP0.PREEDGE (CAP0.PREEDGE, CAP1.PREEDGE (pg 439))	68		8-bit
CAP0.PREFILTER (CAP0.PREFILTER, CAP1.PREFILTER (pg 440))	70		8-bit
CAP0.PRESELECT (CAP0.PRESELECT, CAP1.PRE-SELECT (pg 441))	72		8-bit
CAP0.STATE (CAP0.STATE, CAP1.STATE (pg 443))	74		8-bit
CAP0.T (CAP0.T, CAP1.T (pg 444))	76		32-bit
CAP0.TRIGGER (CAP0.TRIGGER, CAP1.TRIGGER (pg 445))	78		8-bit
CAP1.EDGE	80		8-bit
CAP1.EN	82		8-bit
CAP1.EVENT	84		8-bit

Parameter	Modbus Register Address	Is 64-bit?	Attributes
CAP1.FILTER	86		8-bit
CAP1.MODE	88		8-bit
CAP1.PLFB	90	Yes	64-bit, signed
CAP1.PREEDGE	94		8-bit
CAP1.PREFILTER	96		8-bit
CAP1.PRESELECT	98		8-bit
CAP1.STATE	100		8-bit
CAP1.T	102		32-bit
CAP1.TRIGGER	104		8-bit
CS.DEC (pg 448)	106	Yes	64bit
CS.STATE (pg 450)	110		8-bit
CS.TO (pg 451)	112		32-bit
CS.VTHRESH (pg 452)	114	Yes	low 32-bit word
DIN.ROTARY (PG 457)	116		8-bit
DIN1.INV (DIN1.INV to DIN7.INV (pg 460))	120		8-bit
DIN1.MODE (DIN1.MODE TO DIN24.MODE (pg 461))	122		16-bit
DIN1.PARAM (DIN1.PARAM TO DIN7.PARAM (pg 463))	124	Yes	64-bit, signed
DIN1.STATE (DIN1.STATE TO DIN7.STATE (pg 465))	128		8-bit
DIN2.INV	130		8-bit
DIN2.MODE	132		16-bit
DIN2.PARAM	134	Yes	64-bit, signed
DIN2.STATE	138		8-bit
DIN3.INV	140		8-bit
DIN3.MODE	142		16-bit
DIN3.PARAM	144	Yes	64-bit, signed
DIN3.STATE	148		8-bit
DIN4.INV	150		8-bit
DIN4.MODE	152		16-bit
DIN4.PARAM	154	Yes	64-bit, signed
DIN4.STATE	158		8-bit
DIN5.INV	160		8-bit
DIN5.MODE	162		16-bit
DIN5.PARAM	164	Yes	64-bit, signed
DIN5.STATE	168		8-bit
DIN6.INV	170		8-bit
DIN6.MODE	172		16-bit
DIN6.PARAM	174	Yes	64-bit, signed
DIN6.STATE	178		8-bit
DIN7.INV	180		8-bit
DIN7.MODE	182		16-bit
DIN7.PARAM	184	Yes	64-bit, signed
DIN7.STATE	188		8-bit
DOUT.CTRL (pg 473)	190		8-bit

Parameter	Modbus Register Address	Is 64-bit?	Attributes
DOUT.RELAYMODE (pg 474)	192		8-bit
DOUT.STATES (pg 475)	194		8-bit
DOUT1.MODE (DOUT1.MODE to DOUT19.MODE (pg 476))	196		8-bit
DOUT1.PARAM (DOUT1.PARAM AND DOUT2.PARAM (pg 478))	198	Yes	64-bit, signed
DOUT1.STATE (DOUT1.STATE AND DOUT2.STATE (pg 480))	202		8-bit
DOUT1.STATEU (DOUT1.STATEU AND DOUT2.STATEU (pg 481))	204		8-bit
DOUT2.MODE (DOUT1.MODE to DOUT19.MODE (pg 476))	206		8-bit
DOUT2.PARAM (DOUT1.PARAM AND DOUT2.PARAM (pg 478))	208	Yes	64-bit, signed
DOUT2.STATE (DOUT1.STATE AND DOUT2.STATE (pg 480))	212		8-bit
DOUT2.STATEU (DOUT1.STATEU AND DOUT2.STATEU (pg 481))	214		8-bit
DRV.ACC (PG 489)	216	Yes	64-bit
DRV.ACTIVE (PG 491)	220		8-bit
DRV.CLRFAULTHIST (PG 494)	222		Command
DRV.CLRFAULTS (PG 495)	224		Command
DRV.CMDSOURCE (PG 497)	226		8-bit
DRV.DBILIMIT (pg 500)	228		32-bit
DRV.DEC (PG 501)	230	Yes	64-bit
DRV.DIR (pg 504)	234		8-bit
DRV.DIS (PG 506)	236		Command
DRV.DISMODE (pg 507)	238		8-bit
DRV.DISSOURCES (PG 509)	240		16-bit
DRV.DISTO (pg 511)	242		32-bit
DRV.EMUEDIR (pg 513)	244		8-bit
DRV.EMUEMODE (pg 514)	246		16-bit
DRV.EMUEMTURN (pg 516)	248		32-bit
DRV.EMUERES (pg 518)	250		32-bit
DRV.EMUEZOFFSET (pg 519)	252		16-bit
DRV.EN (PG 520)	254		Command
DRV.ENDEFAULT (pg 521)	256		8-bit
DRV.HANDWHEEL (pg 525)	258		32-bit
DRV.HWENMODE (pg 531)	260		8-bit
DRV.ICONT (PG 532)	262		32-bit, signed
DRV.IPEAK (PG 535)	264		32-bit, signed
DRV.IZERO (pg 536)	266		32-bit
DRV.MOTIONSTAT (PG 541)	268		32-bit
DRV.OPMODE (PG 549)	270		8-bit
DRV.RSTVAR (PG 552)	272		Command

Parameter	Modbus Register Address	Is 64-bit?	Attributes
DRV.STOP (PG 556)	274		Command
DRV.TYPE (pg 559)	276		8-bit
DRV.ZERO (pg 565)	278		8-bit
FB1.BISSBITS (pg 573)	280		8-bit
FB1.ENCRES (PG 575)	282		32-bit
FB1.IDENTIFIED (PG 583)	284		8-bit
FB1.INITSIGNED (pg 584)	286		8-bit, signed
FB1.MECHPOS (PG 585)	288		32-bit
FB1.ORIGIN (pg 588)	294	Yes	64-bit
FB1.PFIND (pg 592)	298		8-bit
FB1.PFINDCMDU (pg 593)	300		32-bit
FB1.POLES (PG 597)	302		16-bit
FB1.PSCALE (pg 598)	304		8-bit
FB1.RESKTR (pg 601)	306		16-bit
FB1.RESREFPHASE (pg 602)	308		32-bit, signed
FB1.SELECT (PG 603)	310		8-bit, signed
FB1.TRACKINGCAL (pg 606)	312		8-bit
FBUS.PARAM01 (FBUS.PARAM1 TO FBUS.PARAM10 (pg 625))	314		32-bit
FBUS.PARAM02	316		32-bit
FBUS.PARAM03	318		32-bit
FBUS.PARAM04	320		32-bit
FBUS.PARAM05	322		32-bit
FBUS.PARAM06	324		32-bit
FBUS.PARAM07	326		32-bit
FBUS.PARAM08	328		32-bit
FBUS.PARAM09	330		32-bit
FBUS.PARAM10	332		32-bit
FBUS.PLLTHRESH (pg 629)	354		16-bit
FBUS.SAMPLEPERIOD (pg 633)	356		8-bit
FBUS.SYNCACT (pg 635)	358		32-bit
FBUS.SYNCDIST (pg 636)	360		32-bit
FBUS.SYNCWND (pg 637)	362		32-bit
FBUS.TYPE (pg 638)	364		8-bit
GEAR.ACCMAX (pg 640)	366	Yes	64-bit
GEAR.DECMAX (pg 641)	370	Yes	64-bit
GEAR.IN (pg 643)	374		16-bit
GEAR.MODE (pg 644)	376		16-bit
GEAR.MOVE (pg 646)	378		Command
GEAR.OUT (pg 647)	380		16-bit, signed
GEAR.VMAX (pg 648)	382	Yes	low 32bit word,
HOME.ACC (pg 663)	384	Yes	64-bit
HOME.AUTOMOVE (pg 665)	388		8-bit

Parameter	Modbus Register Address	Is 64-bit?	Attributes
HOME.DEC (pg 666)	390	Yes	64-bit
HOME.DIR (pg 668)	394		16-bit
HOME.DIST (pg 669)	396	Yes	64-bit, signed
HOME.FEEDRATE (pg 670)	400		16-bit
HOME.IPEAK (pg 671)	402	Yes	64-bit, signed
HOME.MODE (pg 672)	406		16-bit
HOME.MOVE (pg 674)	408		Command
HOME.P (pg 675)	410	Yes	64-bit, signed
HOME.PERRTHRESH (pg 676)	414	Yes	64-bit, signed
HOME.SET (pg 678)	418		Command
HOME.V (pg 679)	420	Yes	Low 32-bit word
HWLS.NEGSTATE (pg 682)	422		8-bit
HWLS.POSSTATE (pg 683)	424		8-bit
IL.BUSFF (pg 686)	426		32-bit, signed
IL.CMD (PG 687)	428		32-bit, signed
IL.CMDU (PG 689)	430		32-bit, signed
IL.FB (PG 691)	432		32-bit, signed
IL.FF (pg 693)	434		32-bit
IL.FOLDFTHRESH (pg 694)	436		32-bit
IL.FOLDFTHRESHU (pg 695)	438		32-bit, signed
IL.FOLDWTHRESH (pg 696)	440		32-bit, signed
IL.FRICTION (pg 697)	442		32-bit
IL.IFOLD (pg 698)	444		32-bit
IL.IUFB (pg 699)	446		32-bit, signed
IL.IVFB (pg 700)	448		32-bit, signed
IL.KACCFF (pg 701)	450		32-bit, signed
IL.KBUSFF (pg 702)	452		32-bit
IL.KP (PG 703)	454		16-bit
IL.KPDRATIO (PG 704)	456		32-bit
IL.KVFF (pg 708)	458		32-bit, signed
IL.LIMITN (PG 709)	460		32-bit, signed
IL.LIMITP (PG 710)	462		32-bit, signed
IL.MFOLDD (PG 711)	464		32-bit
IL.MFOLDR (PG 712)	466		32-bit
IL.MFOLDT (PG 713)	468		32-bit
IL.MIFOLD (PG 716)	470		32-bit
IL.OFFSET (pg 718)	472		32-bit, signed
IL.VCMD (PG 719)	474		16-bit, signed
IL.VUFB (PG 720)	476		16-bit, signed
IL.VVFB (PG 721)	478		16-bit, signed
MOTOR.AUTOSET (pg 742)	480		8-bit
MOTOR.BRAKE (pg 744)	482		8-bit
MOTOR.BRAKERLS (pg 746)	484		8-bit

Parameter	Modbus Register Address	Is 64-bit?	Attributes
MOTOR.CTF0 (pg 748)	486		32-bit
MOTOR.ICONT (pg 749)	488		32-bit
MOTOR.IDDATAVALID (pg 750)	490		8-bit
MOTOR.INERTIA (pg 754)	492		32-bit
MOTOR.IPEAK (pg 755)	494		32-bit
MOTOR.KT (pg 757)	496		32-bit
MOTOR.LQLL (pg 758)	498		32-bit
MOTOR.PHASE (PG 760)	500		16-bit
MOTOR.PITCH (PG 761)	502		32-bit
MOTOR.POLES (PG 762)	504		16-bit
MOTOR.R (PG 763)	506		32-bit
MOTOR.RTYPE (pg 764)	508		8-bit
MOTOR.TBRAKEAPP (PG 765)	510		16-bit
MOTOR.TBRAKERLS (PG 766)	512		16-bit
MOTOR.TBRAKETO (pg 767)	990		32-bit, signed
MOTOR.TEMP (pg 768)	514		32-bit
MOTOR.TEMPFAULT (pg 769)	516		32-bit
MOTOR.TEMPWARN (pg 770)	518		32-bit
MOTOR.TYPE (PG 771)	520		8-bit
MOTOR.VMAX (pg 773)	522		16-bit
MOTOR.VOLTMAX (PG 774)	524		16-bit
MOTOR.VOLTMIN (pg 775)	998		16.bit
MOTOR.VOLTRATED (pg 776)	992		16-bit
MOTOR.VRATED (pg 777)	994	Yes	64-bit, signed
MT.ACC (PG 780)	526	Yes	64-bit
MT.CLEAR (PG 782)	530		16-bit, signed
MT.CNTL (PG 783)	532		32-bit
MT.CONTINUE (PG 786)	534		Command
MT.DEC (PG 787)	536	Yes	64-bit
MT.EMERGMT (PG 789)	540		16-bit, signed
MT.LOAD (PG 792)	542		Command
MT.MOVE (PG 793)	544		Command 16-bit
MT.MTNEXT (pg 794)	546		8-bit
MT.NUM (pg 795)	548		8-bit
MT.P (PG 796)	550	Yes	64-bit, signed
MT.SET (PG 798)	554		Command 8-bit
MT.TNEXT (pg 799)	556		16-bit
MT.NUM (PG 795)	558		8-bit
MT.TPOSWND (pg 802)	560	Yes	64-bit, signed
MT.TVELWND (pg 803)	564		32-bit
MT.V (PG 805)	566	Yes	low 32-bit word
MT.VCMD (pg 807)	568	Yes	low 32-bit word, signed

Parameter	Modbus Register Address	Is 64-bit?	Attributes
PL.CMD (PG 809)	570	Yes	64-bit
PL.ERR (PG 810)	574	Yes	64-bit
PL.ERRMODE (pg 813)	578		8-bit
PL.ERRFTHRESH (pg 811)	580	Yes	64-bit
PL.ERRWTHRESH (pg 815)	584	Yes	64-bit
PL.FB (PG 817)	588	Yes	64-bit, signed
PL.FBSOURCE (pg 818)	592		8-bit
PL.INTINMAX (PG 820)	594	Yes	64-bit
PL.INTOUTMAX (PG 822)	598	Yes	64-bit
PL.KI (PG 824)	602		32-bit
PL.KP (PG 825)	604		32-bit
PL.MODP1 (pg 826)	606	Yes	64-bit, signed
PL.MODP2 (pg 827)	610	Yes	64-bit, signed
PL.MODPDIR (pg 828)	614		8-bit
PL.MODPEN (pg 829)	616		8-bit
PLS.EN (pg 831)	618		16-bit
PLS.MODE (pg 832)	620		16-bit
PLS.P1 (PLS.P1 TO PLS.P8 (pg 833))	622	Yes	64-bit, signed
PLS.P2	626	Yes	64-bit, signed
PLS.P3	630	Yes	64-bit, signed
PLS.P4	634	Yes	64-bit, signed
PLS.P5	638	Yes	64-bit, signed
PLS.P6	642	Yes	64-bit, signed
PLS.P7	646	Yes	64-bit, signed
PLS.P8	650	Yes	64-bit, signed
PLS.RESET (pg 835)	654		16-bit
PLS.STATE (pg 836)	656		16-bit
PLS.T1 (PLS.T1 TO PLS.T8 (pg 837))	658		16-bit
PLS.T2	660		16-bit
PLS.T3	662		16-bit
PLS.T4	664		16-bit
PLS.T5	666		16-bit
PLS.T6	668		16-bit
PLS.T7	670		16-bit
PLS.T8	672		16-bit
PLS.UNITS (pg 839)	674		8-bit
PLS.WIDTH1 (PLS.WIDTH1 TO PLS.WIDTH8 (pg 841))	676	Yes	64-bit, signed
PLS.WIDTH2	680	Yes	64-bit, signed
PLS.WIDTH3	684	Yes	64-bit, signed
PLS.WIDTH4	688	Yes	64-bit, signed
PLS.WIDTH5	692	Yes	64-bit, signed
PLS.WIDTH6	696	Yes	64-bit, signed
PLS.WIDTH7	700	Yes	64-bit, signed

Parameter	Modbus Register Address	Is 64-bit?	Attributes
PLS.WIDTH8	704	Yes	64-bit, signed
REC.ACTIVE (PG 844)	708		8-bit
REC.DONE (PG 846)	710		8-bit
REC.GAP (PG 847)	712		16-bit
REC.NUMPOINTS (PG 848)	714		16-bit
REC.OFF (PG 849)	716		Command
REC.STOPTYPE (pg 857)	718		8-bit
REC.TRIG (PG 858)	720		Command
REC.TRIGPOS (pg 860)	722		8-bit
REC.TRIGSLOPE (PG 863)	726		8-bit
REC.TRIGTYPE (PG 864)	728		8-bit
REC.TRIGVAL (PG 865)	730	Yes	64-bit, signed
REGEN.POWER (PG 867)	734	Yes	64-bit
REGEN.REXT (PG 869)	738		16-bit
REGEN.TEXT (pg 870)	740		32-bit
REGEN.TYPE (PG 872)	742		8-bit, signed
REGEN.WATTEXT (PG 873)	744		16-bit
SM.I1 (pg 879)	746		32-bit, signed
SM.I2 (pg 880)	748		32-bit, signed
SM.MODE (pg 881)	750		16-bit
SM.MOVE (pg 884)	752		Command
SM.T1 (pg 885)	754		16-bit
SM.T2 (pg 886)	756		16-bit
SM.V1 (pg 887)	758	Yes	low 32-bit word, signed
SM.V2 (pg 889)	760	Yes	low 32-bit word, signed
STO.STATE (pg 892)	762		8-bit
SWLS.EN (pg 894)	764		16-bit
SWLS.LIMIT0 (pg 895)	766	Yes	64-bit, signed
SWLS.LIMIT1 (pg 896)	770	Yes	64-bit, signed
SWLS.STATE (pg 897)	774		8-bit
UNIT.ACCLINEAR (PG 902)	776		8-bit
UNIT.ACCROTARY (PG 903)	778		8-bit
UNIT.PIN (PG 905)	780		32-bit
UNIT.PLINEAR (PG 906)	782		8-bit
UNIT.POUT (PG 907)	784		32-bit
UNIT.PROTARY (PG 908)	786		8-bit
UNIT.VLINEAR (PG 909)	788		8-bit
UNIT.VROTARY (PG 910)	790		8-bit
VBUS.CALGAIN	792		32-bit
VBUS.OVFTHRESH (pg 913)	794		16-bit
VBUS.OVWTHRESH (pg 914)	796		16-bit

Parameter	Modbus Register Address	Is 64-bit?	Attributes
VBUS.RMSLIMIT (pg 915)	798		8-bit
VBUS.UVFTHRESH (pg 916)	800		16-bit
VBUS.UVMODE (pg 917)	802		8-bit
VBUS.UVWTHRESH (pg 918)	804		16-bit
VBUS.VALUE (pg 919)	806		32-bit
VL.ARPF1 (VL.ARPF1 TO VL.ARPF4 (pg 921))	808		32-bit
VL.ARPF2	810		32-bit
VL.ARPF3	812		32-bit
VL.ARPF4	814		32-bit
VL.ARPQ1 (VL.ARPQ1 TO VL.ARPQ4 (pg 923))	816		32-bit
VL.ARPQ2	818		32-bit
VL.ARPQ3	820		32-bit
VL.ARPQ4	822		32-bit
VL.ARTYPE1 (VL.ARTYPE1 TO VL.ARTYPE4 (pg 925))	824		8-bit
VL.ARTYPE2	826		8-bit
VL.ARTYPE3	828		8-bit
VL.ARTYPE4	830		8-bit
VL.ARZF1 (VL.ARZF1 TO VL.ARZF4 (pg 926))	832		32-bit
VL.ARZF2	834		32-bit
VL.ARZF3	836		32-bit
VL.ARZF4	838		32-bit
VL.ARZQ1 (VL.ARZQ1 TO VL.ARZQ4 (pg 928))	840		32-bit
VL.ARZQ2	842		32-bit
VL.ARZQ3	844		32-bit
VL.ARZQ4	846		32-bit
VL.BUSFF (pg 930)	848	Yes	low 32-bit word, signed
VL.CMD (PG 931)	850	Yes	low 32-bit word, signed
VL.CMDU (PG 932)	852	Yes	low 32-bit word, signed
VL.ERR (PG 933)	854	Yes	low 32-bit word, signed
VL.FB (PG 934)	856	Yes	low 32-bit word, signed
VL.FBFILTER (pg 935)	858	Yes	low 32-bit word, signed
VL.FBSOURCE (pg 936)	860		8-bit
VL.FF (pg 938)	862	Yes	low 32-bit word, signed
VL.GENMODE (PG 939)	864		16-bit
VL.KBUSFF (pg 940)	866		32-bit
VL.KI (pg 941)	868		32-bit
VL.KO (pg 943)	870		32-bit

Parameter	Modbus Register Address	Is 64-bit?	Attributes
VL.KP (pg 944)	872		32-bit
VL.KVFF (pg 946)	874		32-bit
VL.LIMITN (PG 947)	876	Yes	low 32-bit word, signed
VL.LIMITP (PG 949)	878	Yes	low 32-bit word
VL.LMJR (pg 951)	880		32-bit
VL.MODEL (pg 952)	882	Yes	low 32-bit word, signed
VL.OBSBW (pg 953)	884		32-bit
VL.OBSMODE (pg 954)	886		32-bit
VL.THRESH (PG 955)	888	Yes	low 32-bit word, signed
WS.ARM (pg 958)	890		Command
WS.DISTMAX (pg 963)	892	Yes	64-bit, signed
WS.DISTMIN (pg 964)	896	Yes	64-bit, signed
WS.IMAX (pg 966)	900		32-bit, signed
WS.MODE (pg 967)	902		8-bit
WS.NUMLOOPS (pg 968)	904		8-bit
WS.STATE (pg 969)	906		8-bit
WS.T (pg 970)	908		16-bit
WS.TDELAY1 (pg 971)	910		16-bit
WS.TDELAY2 (pg 972)	912		16-bit
WS.TDELAY3 (pg 973)	914		16-bit
WS.VTHRESH (pg 976)	916	Yes	low 32-bit word, signed
DIN1.FILTER (DIN1.FILTER TO DIN7.FILTER (pg 459))	918		16-bit
DIN2.FILTER	920		16-bit
DIN3.FILTER	922		16-bit
DIN4.FILTER	924		16-bit
DIN5.FILTER	926		16-bit
DIN6.FILTER	928		16-bit
DIN7.FILTER	930		16-bit
FB1.HALLSTATEU (pg 580)	932		8-bit
FB1.HALLSTATEV (pg 581)	934		8-bit
FB1.HALLSTATEW (pg 582)	936		8-bit
DRV.NVSAVE (pg 547)	938		Command
MODBUS.DIO	940		32-bit
MODBUS.DRV	942		32-bit
MODBUS.DRVSTAT	944		32-bit
MODBUS.HOME	946		32-bit
MODBUS.MOTOR	948		32-bit
MODBUS.MT	950		16-bit
MODBUS.SM	952		32-bit

Parameter	Modbus Register Address	Is 64-bit?	Attributes
DRV.FAULT1 (DRV.FAULT1 to DRV.FAULT10 (pg 523))	954		16-bit
DRV.FAULT2	956		16-bit
DRV.FAULT3	958		16-bit
DRV.FAULT4	960		16-bit
DRV.FAULT5	962		16-bit
DRV.FAULT6	964		16-bit
DRV.FAULT7	966		16-bit
DRV.FAULT8	968		16-bit
DRV.FAULT9	970		16-bit
DRV.FAULT10	972		16-bit
MODBUS.PIN (pg 736)	974		32-bit
MODBUS.POUT (pg 737)	976		32-bit
MODBUS.PSCALE (pg 738)	978		16-bit
MOTOR.HFPHASEREAD	982		16-bit
FB2.ENCRES (pg 611)	984		32-bit
FB2.MODE (pg 612)	986		16-bit
"FB2.SOURCE " (➜ p. 617)	988		16-bit
MOTOR.TBRAKETO (pg 767)	990		32-bit, signed
MODBUS.MSGLOG	992		8-bit
USER.INT1	994		32-bit, signed
USER.INT2	996		32-bit, signed
USER.INT3	998		32-bit, signed
USER.INT4	1000		32-bit, signed
USER.INT5	1002		32-bit, signed
USER.INT6	1004		32-bit, signed
USER.INT7	1006		32-bit, signed
USER.INT8	1008		32-bit, signed
USER.INT9	1010		32-bit, signed
USER.INT10	1012		32-bit, signed
USER.INT11	1014		32-bit, signed
USER.INT12	1016		32-bit, signed
USER.INT13	1018		32-bit, signed
USER.INT14	1020		32-bit, signed
USER.INT15	1022		32-bit, signed
USER.INT16	1024		32-bit, signed
USER.INT17	1026		32-bit, signed
USER.INT18	1028		32-bit, signed
USER.INT19	1030		32-bit, signed
USER.INT20	1032		32-bit, signed
USER.INT21	1034		32-bit, signed
USER.INT22	1036		32-bit, signed
USER.INT23	1038		32-bit, signed
USER.INT24	1040		32-bit, signed

Parameter	Modbus Register Address	Is 64-bit?	Attributes
DRV.NVCHECK (pg 544)	1042	Yes	low 32-bit word
MODBUS.SCALING (pg 739)	1048		8-bit
DRV.EMUEPULSEWIDTH (pg 517)	1050		32-bit
DRV.EMUECHECKSPEED (pg 512)	1052		8-bit
DRV.HWENABLE (pg 529)	1054		8-bit
DRV.SWENABLE	1056		8-bit
DRV.TIME (pg 558)	1058		32-bit
EGEAR.ACCLIMIT	1060	Yes	low 32-bit word
EGEAR.DECLIMIT	1062	Yes	low 32-bit word
EGEAR.ERROR	1064	Yes	low 32-bit word
EGEAR.LOCK	1066		8-bit
EGEAR.ON	1068		8-bit
EGEAR.PULSESIN	1070		16-bit
EGEAR.PULSEOUT	1072		16-bit signed
EGEAR.RATIO	1074		32-bit
EGEAR.TYPE	1076		8-bit
EXTENCODER.FREQ	1078		32-bit
EXTENCODER.POSITION	1080	Yes	64-bit signed
EXTENCODER.POSMODULO	1084	Yes	64-bit
MOVE.ACC	1088	Yes	64-bit
MOVE.DEC	1092	Yes	64-bit
MOVE.DEC	1096		32-bit
MOVE.GOABS	1098		Command
MOVE.GOABSREG	1100		Command
MOVE.GOHOME	1102		Command
MOVE.GORELREG	1104		Command
MOVE.GOREL	1106		Command
MOVE.GOPUDATE	1108		Command
MOVE.GOVEL	1110		Command
MOVE.INPOSITION	1112		32-bit
MOVE.INPOSLIMIT	1114	Yes	64-bit signed
MOVE.MOVING	1118		32-bit
MOVE.POSCOMMAND	1120	Yes	64-bit signed
MOVE.REGOFFSET	1130	Yes	64-bit signed
MOVE.RELATIVEDIST	1134	Yes	64-bit signed
MOVE.RUNSPEED	1138	Yes	64-bit
MOVE.SCURVETIME	1142		32-bit
MOVE.ABORT	1144		Command
MOVE.TARGETPOS	1146	Yes	64-bit
MOVE.VCMD	1150		32-bit
VM.AUTOSTART	1152		32-Bit
VM.RESTART	1154		Command
VM.START	1156		Command

Parameter	Modbus Register Address	Is 64-bit?	Attributes
VM.STATE	1158		8-bit
VM.STOP	1160		Command
VM.ERR	1162		32-bit
WHEN.FB1MECHPOS	1164		32-bit
WHEN.FB3P	1166		64-bit
WHEN.DRVHANDWHEEL	1170		32-bit
WHEN.DRVTIME	1172		32-bit
WHEN.PLCMD	1174	Yes	64-bit
WHEN.PLFB	1178	Yes	64-bit signed
MOVE.DWELLTIME	1182		32-bit
IL.MI2T (pg 714)	1184		16-bit
AIN.DEADBANDMODE (pg 368)	1186		16-bit
AIN.MODE (pg 371)	1188		8-bit
DIO10.DIR (DIO9.DIR to DIO11.DIR (pg 471))	1190		8-bit
DIO10.INV (DIO9.INV to DIO11.INV (pg 470))	1192		8-bit
DIO11.DIR	1194		8-bit
DIO11.INV	1196		8-bit
DIO9.DIR	1198		8-bit
DIO9.INV	1200		8-bit
FAULT130.ACTION (FAULTx.ACTION (pg 571))	1202		8-bit
FAULT131.ACTION	1204		8-bit
FAULT132.ACTION	1206		8-bit
FAULT134.ACTION	1208		8-bit
FAULTx.ACTION (pg 571)	1210		8-bit
IP.MODE (pg 725)	1212		16-bit
LOAD.INERTIA (pg 731)	1214		32-bit
MOTOR.KE (pg 756)	1216		32-bit
VBUS.HALFVOLT (pg 912)	1218		8-bit
FB2.DIR (pg 614)	1220		8-bit
FAULT451.ACTION (FAULTx.ACTION (pg 571))	1222		8 bit
DRV.HWENDELAY (pg 530)	1224		8-bit
DRV.HANDWHEELSRC (pg 526)	1226		8-bit
IL.KPLOOKUPINDEX (pg 705)	1228		16-bit
IL.KPLOOKUPVALUE (pg 706)	1230		32-bit
MOTOR.BRAKEIMM (pg 745)	1232		8-bit
AIN2.CUTOFF (pg 380)	1234		32-bit
AIN2.DEADBAND (pg 381)	1236		16-bit
AIN2.DEADBANDMODE (pg 382)	1238		16-bit
AIN2.MODE (pg 383)	1242		8-bit
AIN2.OFFSET (pg 384)	1244		16-bit, signed
AIN2.VALUE (pg 385)	1250		16-bit
AIN2.ZERO (pg 386)	1258		Command
AIO.ISCALE (pg 388)	1260		32-bit

Parameter	Modbus Register Address	Is 64-bit?	Attributes
AIO.PSCALE (pg 389)	1262	Yes	64-bit
AIO.VSCALE (pg 391)	1266	Yes	64-bit
AOUT.CUTOFF (pg 394)	1270		32-bit
AOUT2.CUTOFF (pg 406)	1272		32-bit
AOUT2.MODE (pg 407)	1276		16-bit
AOUT2.OFFSET (pg 408)	1278		16-bit, signed
AOUT2.VALUE (pg 409)	1284	Yes	64-bit, signed
BODE.IFLIMIT (pg 415)	1296		32-bit, signed
BODE.IFTHRESH (pg 416)	1298		32-bit, signed
BODE.VFLIMIT (pg 427)	1300		32-bit, signed
BODE.VFTHRESH (pg 428)	1302	Yes	64-bit, signed
DIN10.STATE (DIN9.STATE to DIN11.STATE (pg 466))	1306		8-bit
DIN11.STATE	1308		8-bit
DIN21.FILTER (DIN21.FILTER to DIN32.FILTER (pg 467))	1310		16-bit
DIN21.STATE (DIN21.STATE to DIN32.STATE (pg 468))	1320		8-bit
DIN22.FILTER	1322		16-bit
DIN22.STATE	1332		8-bit
DIN23.FILTER	1334		16-bit
DIN23.STATE	1344		8-bit
DIN24.FILTER	1346		16-bit
DIN24.STATE	1356		8-bit
DIN25.FILTER	1358		16-bit
DIN25.STATE	1368		8-bit
DIN26.FILTER	1370		16-bit
DIN26.STATE	1380		8-bit
DIN27.FILTER	1382		16-bit
DIN27.STATE	1392		8-bit
DIN28.FILTER	1394		16-bit
DIN28.STATE	1404		8-bit
DIN29.FILTER	1406		16-bit
DIN29.STATE	1416		8-bit
DIN30.FILTER	1418		16-bit
DIN30.STATE	1428		8-bit
DIN31.FILTER	1430		16-bit
DIN31.STATE	1440		8-bit
DIN32.FILTER	1442		16-bit
DIN32.STATE	1452		8-bit
DIN9.STATE (DIN9.STATE to DIN11.STATE (pg 466))	1454		8-bit
DOUT10.STATE (DOUT9.STATE to DOUT11.STATE (pg 482))	1456		8-bit
DOUT10.STATEU (DOUT9.STATEU to DOUT11.STATEU (pg 483))	1458		8-bit
DOUT11.STATE	1460		8-bit
DOUT11.STATEU	1462		8-bit

Parameter	Modbus Register Address	Is 64-bit?	Attributes
DOUT21.STATE (DOUT21.STATE to DOUT32.STATE (pg 485))	1470		8-bit
DOUT21.STATEU (DOUT21.STATEU to DOUT32.STATEU (pg 486))	1472		8-bit
DOUT22.STATE	1480		8-bit
DOUT22.STATEU	1482		8-bit
DOUT23.STATE	1490		8-bit
DOUT23.STATEU	1492		8-bit
DOUT24.STATE	1500		8-bit
DOUT24.STATEU	1502		8-bit
DOUT25.STATE	1510		8-bit
DOUT25.STATEU	1512		8-bit
DOUT26.STATE	1520		8-bit
DOUT26.STATEU	1522		8-bit
DOUT27.STATE	1530		8-bit
DOUT27.STATEU	1532		8-bit
DOUT28.STATE	1540		8-bit
DOUT28.STATEU	1542		8-bit
DOUT29.STATE	1550		8-bit
DOUT29.STATEU	1552		8-bit
DOUT30.STATE	1560		8-bit
DOUT30.STATEU	1562		8-bit
DOUT9.STATE (DIN9.STATE to DIN11.STATE (pg 466))	1564		8-bit
DOUT9.STATEU (DOUT9.STATEU to DOUT11.STATEU (pg 483))	1566		8-bit
DRV.BLINKDISPLAY (pg 492)	1568		Command
DRV.CMDDELAY (pg 496)	1572		
DRV.NVLOAD (pg 546)	1576		Command
DRV.RUNTIME (pg 553)	1578		
DRV.SETUPREQBITS (pg 554)	1580		32-bit
DRV.WARNING1 to DRV.WARNING10 (pg 563)	1582		16-bit
DRV.WARNING1 to DRV.WARNING10 (pg 563)	1584		16-bit
DRV.WARNING1 to DRV.WARNING10 (pg 563)	1586		16-bit
EIP.POSUNIT (pg 568)	1590		32-bit
EIP.PROFUNIT (pg 569)	1592		32-bit
FAULT139.ACTION (FAULTx.ACTION (pg 571))	1594		8-bit
FB1.P (pg 590)	1610	Yes	64-bit, signed
FB1.PDIR (pg 591)	1614		8-bit
FB1.PIN	1616		32-bit
FB1.POFFSET (pg 596)	1618	Yes	64-bit, signed
FB1.POUT	1622		32-bit
FB1.PUNIT (pg 600)	1624		32-bit
FB2.P (pg 613)	1632	Yes	64-bit, signed
FB2.PIN	1636		32-bit

Parameter	Modbus Register Address	Is 64-bit?	Attributes
FB2.POFFSET (pg 615)	1638	Yes	64-bit, signed
FB2.POUT	1642		32-bit
FB2.PUNIT (pg 616)	1644		32-bit
FB3.P (pg 620)	1646	Yes	64-bit, signed
FB3.PDIR (pg 621)	1650		8-bit
FB3.PIN	1652		32-bit
FB3.POFFSET (pg 622)	1654	Yes	64-bit, signed
FB3.POUT	1658		32-bit
FB3.PUNIT (pg 623)	1660		32-bit
IL.DIFOLD (pg 690)	1666		32-bit
IL.MI2TWTHRESH (pg 715)	1668		8-bit
IL.MIMODE (pg 717)	1670		8-bit
IP.RESET (pg 727)	1672		Command
MOTOR.VOLTMIN (pg 775)	1674		16-bit
MOTOR.VOLTRATED (pg 776)	1676		16-bit
MOTOR.VRATED (pg 777)	1678	Yes	64-bit, signed
MT.HOMEREQUIRE (pg 790)	1682		8-bit
SD.LOAD (pg 875)	1684		Command
SD.SAVE (pg 876)	1686		Command
SD.STATUS (pg 877)	1688		8-bit
VL.FBUNFILTERED (pg 937)	1690	Yes	64-bit, signed
WS.DISARM (pg 962)	1694		Command
WS.FREQ (pg 965)	1696		32-bit
WS.TDELAY4	1698		16-bit
WS.CHECKT (pg 960)	1700		16-bit
WS.CHECKV (pg 961)	1702	Yes	64-bit, signed
WS.TSTANDSTILL (pg 975)	1720		16-bit
WS.TIRAMP (pg 974)	1722		16-bit
FB1.PMTSAVEEN	1724		8-bit
FB1.PMTBITS (pg 594)	1726		8-bit
MOTOR.IMTR (pg 752)	1728		16-bit
IL.FBSOURCE (pg 692)	1730		8-bit
MOTOR.IMID (pg 751)	1732		32-bit
WS.CHECKMODE (pg 959)	1734		8-bit
REGEN.POWERFILTERED (pg 868)	1736	Yes	64-bit
FBUS.PROTECTION (pg 630)	1742		8-bit
FBUS.BLOCKING	1744		8-bit
FBUS.STATE (pg 634)	1746		8-bit, signed
TEMP.CONTROL (pg 899)	1748		16-bit, signed
TEMP.POWER1 (TEMP.POWER1 to TEMP.POWER3 (pg 900))	1750		16-bit,signed
TEMP.POWER2	1752		16-bit,signed
TEMP.POWER3	1754		16-bit,signed

Parameter	Modbus Register Address	Is 64-bit?	Attributes
MODBUS.ERRORMODE (pg 734)	1756		8-bit
MODBUS.CLRERRORS (pg 733)	1758		Command
IL.CMDACC (pg 688)	1760	Yes	64-bit, signed

This page intentionally left blank.

Modbus 64-bit Parameters to 32-bit Mapping

Parameter	Address	64-bit	Attributes
AIN.PSCALE_32	2000	Yes	low 32-bit word, signed
AOUT.PSCALE_32	2002	Yes	low 32-bit word
AOUT.VALUE_32	2004	Yes	low 32-bit word, signed
AOUT.VALUEU_32	2006	Yes	low 32-bit word, signed
CAP0.PLFB_32	2008	Yes	low 32-bit word, signed
CAP1.PLFB_32	2010	Yes	low 32-bit word, signed
CS.DEC_32	2012	Yes	low 32-bit word
DIN1.PARAM_32	2014	Yes	low 32-bit word, signed
DIN2.PARAM_32	2016	Yes	low 32-bit word, signed
DIN3.PARAM_32	2018	Yes	low 32-bit word, signed
DIN4.PARAM_32	2020	Yes	low 32-bit word, signed
DIN5.PARAM_32	2022	Yes	low 32-bit word, signed
DIN6.PARAM_32	2024	Yes	low 32-bit word, signed
DIN7.PARAM_32	2026	Yes	low 32-bit word, signed
DOUT1.PARAM_32	2028	Yes	low 32-bit word, signed
DOUT2.PARAM_32	2030	Yes	low 32-bit word, signed
DRV.ACC_32	2032	Yes	low 32-bit word
DRV.DEC_32	2034	Yes	low 32-bit word
FB1.OFFSET_32	2036	Yes	low 32-bit word, signed
FB1.ORIGIN_32	2038	Yes	low 32-bit word
GEAR.ACCMAX_32	2040	Yes	low 32-bit word
GEAR.DECMAX_32	2042	Yes	low 32-bit word
HOME.ACC_32	2044	Yes	low 32-bit word
HOME.DEC_32	2046	Yes	low 32-bit word
HOME.DIST_32	2048	Yes	low 32-bit word, signed
HOME.IPEAK_32	2050	Yes	low 32-bit word, signed
HOME.P_32	2052	Yes	low 32-bit word, signed
HOME.PERRTHRESH_32	2054	Yes	low 32-bit word, signed
MOTOR.VRATED_32	2126	Yes	low 32-bit word, signed
MT.ACC_32	2056	Yes	low 32-bit word
MT.DEC_32	2058	Yes	low 32-bit word
MT.P_32	2060	Yes	low 32-bit word, signed
MT.TPOSWND_32	2062	Yes	low 32-bit word, signed
PL.CMD_32	2064	Yes	low 32-bit word
PL.ERR_32	2066	Yes	low 32-bit word
PL.ERRFTHRESH_32	2068	Yes	low 32-bit word
PL.ERRWTHRESH_32	2070	Yes	low 32-bit word
PL.FB_32	2072	Yes	low 32-bit word, signed
PL.INTINMAX_32	2074	Yes	low 32-bit word
PL.INTOUTMAX_32	2076	Yes	low 32-bit word
PL.MODP1_32	2078	Yes	low 32-bit word, signed

Parameter	Address	64-bit	Attributes
PL.MODP2_32	2080	Yes	low 32-bit word, signed
PLS.P1_32	2082	Yes	low 32-bit word, signed
PLS.P2_32	2084	Yes	low 32-bit word, signed
PLS.P3_32	2086	Yes	low 32-bit word, signed
PLS.P4_32	2088	Yes	low 32-bit word, signed
PLS.P5_32	2090	Yes	low 32-bit word, signed
PLS.P6_32	2092	Yes	low 32-bit word, signed
PLS.P7_32	2094	Yes	low 32-bit word, signed
PLS.P8_32	2096	Yes	low 32-bit word, signed
PLS.WIDTH1_32	2098	Yes	low 32-bit word, signed
PLS.WIDTH2_32	2100	Yes	low 32-bit word, signed
PLS.WIDTH3_32	2102	Yes	low 32-bit word, signed
PLS.WIDTH4_32	2104	Yes	low 32-bit word, signed
PLS.WIDTH5_32	2106	Yes	low 32-bit word, signed
PLS.WIDTH6_32	2108	Yes	low 32-bit word, signed
PLS.WIDTH7_32	2110	Yes	low 32-bit word, signed
PLS.WIDTH8_32	2112	Yes	low 32-bit word, signed
REC.TRIGVAL_32	2114	Yes	low 32-bit word, signed
REGEN.POWER_32	2116	Yes	low 32-bit word
SWLS.LIMIT0_32	2118	Yes	low 32-bit word, signed
SWLS.LIMIT1_32	2120	Yes	low 32-bit word, signed
WS.DISTMAX_32	2122	Yes	low 32-bit word, signed
WS.DISTMIN_32	2124	Yes	low 32-bit word, signed
MOTOR.VRATED_32	2126	Yes	low 32-bit word, signed
EXTENCODER.POSITION_32	2128	Yes	low 32-bit word, signed
EXTENCODER.POSMODULO_32	2130	Yes	low 32-bit word
MOVE.ACC_32	2132	Yes	low 32-bit word
MOVE.DEC_32	2134	Yes	low 32-bit word
MOVE.INPOSLIMIT_32	2136	Yes	low 32-bit word, signed
MOVE.POSCOMMAND_32	2138	Yes	low 32-bit word, signed
MOVE.REGOFFSET_32	2142	Yes	low 32-bit word, signed
MOVE.RELATIVEDIST_32	2144	Yes	low 32-bit word, signed
MOVE.RUNSPEED_32	2146	Yes	low 32-bit word
MOVE.TARGETPOS_32	2148	Yes	low 32-bit word, signed
WHEN.FB3P_32	2150	Yes	low 32-bit word
WHEN.PLCMD_32	2152	Yes	low 32-bit word
WHEN.PLFB_32	2154	Yes	low 32-bit word, signed
AIN2.VSCALE_32	2158	Yes	low 32-bit word
AOUT2.VSCALE_32	2160	Yes	low 32-bit word
WS.CHECKV_32	2162	Yes	low 32-bit word, signed
WS.CHECKT_32	2164	No	16-bit

Modbus Errors

General Functionality

Up to 125 modbus access errors are stored in the modbus error list (MODBUS.ERRORS (pg 735)). If more errors occur, the oldest error is dropped and the new error is stored at the end of this list. All modbus errors are stored regardless of the state of the error mode (MODBUS.ERRORMODE (pg 734)).

A block read/write request will always be fully executed. If a register access causes an error, the error will be added to the error list and processing will continue with the next register.

If MODBUS.ERRORMODE is set to 1, the drive will not return an error response. Check MODBUS.ERRORS to verify that the last request was successfully completed. If it returns 0, the request was successful.

The recommended procedure for working in MODBUS.ERRORMODE = 1 is:

1. Clear the error list (MODBUS.CLRERRORS (pg 733))
2. Send the request
3. Check MODBUS.ERRORS. If 0, the request was successful. If non-zero, handle the recorded errors.
4. Repeat from step 1 as necessary.

Parameter Descriptions

24.0.5 MODBUS.ERRORMODE

Enables or disables error response messages.

0: Send error response messages (default)

1: Do not send error response messages

See full description at MODBUS.ERRORMODE (pg 734).

24.0.6 MODBUS.ERRORS

Lists up to 125 Modbus errors. Each entry contains the Modbus address and the error code of the failed Modbus request.

See full description at MODBUS.ERRORS (pg 735).

24.0.7 MODBUS.CLRERRORS

Clears all errors stored in MODBUS.ERRORS.

See full description at MODBUS.CLRERRORS (pg 733).

Access Through Telnet

Telnet or the WorkBench terminal will display the entire list:

```
-->MODBUS.ERRORS
Error count: 2
[00384]   6 : Argument lower than minimum.
[00390] 109 : Invalid Register address.
```

Access Through Modbus

In Modbus the error list starts at register address 0x1102 (4354).

Modbus Address	Description
4354	Error count
4355	Error 1 : Register Address
4356	Error 1: Error Code
4357	Error 2 : Register Address
4358	Error 2 : Error Code
...	...

Modbus Address	Description
4603	Error 125 : Register Address
4604	Error 125 : Error Code

Modbus Error Descriptions

Error Code	Descriptions
0001	Unknown Fault!
0002	Data Not Found
0003	Data value is invalid.
0004	Argument bigger than maximum.
0005	Too many characters in command.
0006	Argument lower than minimum.
0007	Command is read-only.
0008	Parameter is temporarily write protected.
0009	Argument is out of data-range.
0010	Wrong operation mode for command.
0011	Drive enabled! Disable Drive first.
0012	Drive disabled! Enable Drive first.
0013	Data must be even.
0014	Data must be odd.
0015	Axis is not homed.
0016	Motor is currently in motion.
0017	EEPROM bad or does not exist.
0018	Unknown board.
0019	EEPROM is busy.
0020	Fault exists. Correct fault condition and then clear faults.
0021	Can not clear faults. Issue SW or HW disable first.
0022	Reserved
0023	Invalid motor/feedback poles ratio.
0024	All recorder channels empty.
0025	Process already active.
0026	Function not available for the active command source.
0027	Inappropriate Bode-plot mode for this function
0028	Invalid sine-sweep bode plot mode.
0029	Bode-plot start frequency >= end frequency.
0030	Function not available while active disable is in process.
0031	Iu / Iv offset calculation not finished.
0032	Not enough memory available.
0033	Can not save to EEPROM while in-rush relay is closed.
0034	Buffer overflow.
0035	Invalid motion task parameter.
0036	Invalid motion task number.
0037	Invalid motion task velocity.
0038	Invalid motion task acceleration or deceleration.
0039	Invalid motion task customer profile table number.
0040	Invalid motion task following number.

Error Code	Descriptions
0041	Motion task is not initialized.
0042	Motion task is not interruptible due to the control-word setting.
0043	Motion task must be triggered while another motion task is active.
0044	Motion task target position is out of modulo range.
0045	Motion task target position is beyond the software limit switches.
0046	Test mode is off.
0047	Can not change digital input mode. Issue SW or HW disable first.
0048	Internal Drive procedure active:Active Disable
0049	Invalid interpolation timebase.
0050	Invalid modulo-position setting.
0051	Usage: DRV.HELP prefix.suffix.
0052	Access denied.
0053	Reserved
0054	Unknown Encoder Type.
0055	Wake and Shake is active!
0056	Reserved
0057	Reserved
0058	Reserved
0059	Reserved
0060	Invalid index (out-of-range)!
0061	Not allowed to reset IP while Drive is Enabled!
0062	Dynamic mapping is disabled!
0063	No interrupt handler defined!
0064	Cannot execute when program is running.
0065	Reserved
0066	Command is password protected.
0067	Too many cams defined
0068	Too many cam points defined.
0069	Need to end creating previous cam first.
0070	Need to declare a cam first.
0071	Too many points added to cam table.
0072	First master position must be 0.
0073	Cannot add point without creating cam.
0074	Reserved
0075	Cannot end cam without creating cam.
0076	Master position delta must be greater than 1.
0077	Cannot create an already active cam.
0078	Cam not defined.
0079	Cam not available.
0080	Cannot activate cam with MOVE.RUNSPEED = 0.
0081	Master position outside cam table.
0082	SD Card is not inserted into drive.
0083	SD Card is write protected.
0084	SD Card hardware not installed.

Error Code	Descriptions
0085	Command was not found.
0086	Wrong argument for parameter.
0087	Parameter not record-able.
0088	No parameters allowed for command.
0089	No negative values allowed.
0090	No float allowed.
0091	Wrong number of arguments for parameter.
0092	Not a trigger parameter.
0093	Command is password protected.
0094	Fieldbus is not active.
0095	File not found on SD Card.
0096	File error trying to access the SD Card.
0097	File system error accessing the SD card.
0098	A parameter file value could not be set in the drive.
0099	There was an error writing to a file on the SD Card.
0100	SD Card read/write in progress.
0101	There was an error accessing the BASIC binary file.
0102	BASIC program missing or invalid.
0103	Invalid Feedback Type.
0104	Command not supported on this motor type
0105	Feedback memory access created an error.
0106	Command is blocked while fieldbus is operational.
0107	Cannot execute when FB2.ENCRES = 0.
0108	Unsupported data type.
0109	Invalid register address.
0110	Invalid block size.
0111	Invalid byte count.
0112	Block access no supported.
0113	Invalid access.
0114	Invalid mapping address.

25 Appendix A - Parameter and Command Reference Guide

25.1 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 WorkBench terminal screen. The use of these parameters and commands to perform various drive functions is detailed in related sections of the AKD User Guide.

Drive parameter and command categories include the following:

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: • 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 or Modbus.	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
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

25.1.1 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

Abbreviation	Term
NV	Nonvolatile
P	Position, Proportional, Positive
RLS	Release
R	Resistance
STATE	Status, State, Stat
THRESH	Threshold
T	Time
TMAX	Timeout
U	User
V	Velocity, Volt
W	Warning

25.1.2 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 Input (AIN)		
AIN.CUTOFF (pg 365)	NV	Sets the analog input low-pass filter cutoff frequency.
AIN.DEADBAND (pg 366)	NV	Sets the analog input signal deadband.
AIN.DEADBANDMODE (pg 368)	NV	Sets the analog input deadband mode.
AIN.ISCALE (pg 370)	NV	Sets the analog current scale factor.
AIN.MODE (pg 371)	NV	Analog input mode
AIN.OFFSET (pg 372)	NV	Sets the analog input offset.
AIN.PSCALE (pg 373)	NV	Sets the analog position scale factor.
AIN.VALUE (pg 375)	R/O	Reads the value of the analog input signal.
AIN.VSCALE (pg 376)	NV	Sets analog velocity scale factor.
AIN.ZERO (pg 378)	Command	Zeroes the analog input signal.
Analog Input 2 (AIN2)		
AIN2.CUTOFF (pg 380)	NV	Sets the analog input 2 low-pass filter cutoff frequency.
AIN2.DEADBAND (pg 381)	NV	Sets the analog input 2 signal deadband.
AIN2.DEADBANDMODE (pg 382)	NV	Sets the analog input 2 deadband mode.

Parameter or Command	Type	Description
AIN2.MODE (pg 383)	NV	Analog input 2 mode
AIN2.OFFSET (pg 384)	NV	Sets the analog input 2 offset.
AIN2.VALUE (pg 385)	R/O	Reads the value of the analog input 2 signal.
AIN2.ZERO (pg 386)	Command	Zeroes the analog input 2 signal.
Analog Input/Output (AIO)		
AIO.ISCALE (pg 388)	NV	Sets the analog current scale factor.
AIO.VSCALE (pg 391)	NV	Sets velocity scale factor.
AIO.PSCALE (pg 389)	NV	Sets position scale factor.
Analog Output (AOUT)		
AOUT.CUTOFF (pg 394)	NV	Sets the analog output low-pass filter cutoff frequency.
AOUT.ISCALE (pg 395)	NV	Sets the analog current scale factor.
AOUT.MODE (pg 396)	NV	Sets the analog output mode.
AOUT.OFFSET (pg 398)	NV	Sets the analog output offset.
AOUT.PSCALE (pg 399)	NV	Sets the analog position scale factor.
AOUT.VALUE (pg 401)	NV	Reads the analog output value.
AOUT.VALUEU (pg 402)	R/W	Sets the analog output value.
AOUT.VSCALE (pg 403)	NV	Sets the velocity scale factor for analog output.
Analog Output 2 (AOUT2)		
AOUT2.CUTOFF (pg 406)	NV	Sets the analog output 2 low-pass filter cutoff frequency.
AOUT2.MODE (pg 407)	NV	Sets the analog output 2 mode.
AOUT2.OFFSET (pg 408)	NV	Sets the analog output 2 offset.
AOUT2.VALUE (pg 409)	NV	Reads the analog output 2 value.
AOUT.VALUEU (pg 410)	R/W	Sets the analog output 2 value.
Bode plot (BODE)		
BODE.EXCITEGAP (pg 412)	R/W	Controls how often the excitation is updated.
BODE.FREQ (pg 413)	R/W	Sets the frequency of the sine excitation source.
BODE.IAMP (pg 414)	R/W	Sets current command value used during the Bode procedure.
BODE.IFLIMIT (pg 415)	R/W	Sets the current fault duration limit in seconds for the BODE.MODE 5 stability test.
BODE.IFTHRESH (pg 416)	R/W	Sets the current fault threshold for the BODE.MODE 5 stability test.
BODE.INJECTPOINT (pg 417)	R/W	Sets whether the excitation uses current or velocity excitation type.
BODE.MODE (pg 418)	R/W	Sets the mode of the excitation.
BODE.MODETIMER (pg 422)	R/W	Sets the watchdog timer of the excitation.
BODE.PRBDEPTH (pg 424)	R/W	Sets the length of the PRB signal before it repeats.
BODE.VAMP (pg 425)	R/W	Sets the amplitude of the excitation when in velocity mode.
BODE.VFLIMIT (pg 427)	R/W	Sets the velocity fault duration limit (seconds) for the BODE.MODE 5 stability test
BODE.VFTHRESH (pg 428)	R/W	Sets the current fault threshold for the BODE.MODE 5 stability test.
Capture (CAP)		

Parameter or Command	Type	Description
CAP0.EDGE, CAP1.EDGE (pg 431)	NV	Selects the capture edge.
CAP0.EN, CAP1.EN (pg 432)	NV	Enables or disables the related capture engine.
CAP0.EVENT, CAP1.EVENT (pg 433)	NV	Controls the precondition logic.
CAP0.FILTER, CAP1.FILTER (pg 436)	R/W	Controls the precondition logic.
CAP0.MODE, CAP1.MODE (pg 437)	NV	Selects the captured value.
CAP0.PLFB, CAP1.PLFB (pg 438)	R/O	Reads captured position value.
CAP0.PREEDGE, CAP1.PREEDGE (pg 439)	NV	Selects the capture precondition edge.
CAP0.PREFILTER, CAP1.PREFILTER (pg 440)	NV	Sets the filter for the precondition input source.
CAP0.PRESELECT, CAP1.PRESELECT (pg 441)	NV	Sets the precondition trigger.
CAP0.STATE, CAP1.STATE (pg 443)	R/O	Indicates whether or not trigger source was captured.
CAP0.T, CAP1.T (pg 444)	R/O	Reads time capture (if time capture was configured).
CAP0.TRIGGER, CAP1.TRIGGER (pg 445)	NV	Specifies the trigger source for the position capture.
Central Power Supply (CPS)		
CPS.STOSTATE (pg 1)	R/O	Returns the status of the safe torque signal for each of a CPS device's strings.
Controlled Stop (CS)		
CS.DEC (pg 448)	NV	Sets the deceleration value for the controlled stop process.
CS.STATE (pg 450)	NV	Returns the internal status of the controlled stop process.
CS.TO (pg 451)	NV	Sets the time value for the drive velocity to be within CS.VTHRESH (pg 450).
CS.VTHRESH (pg 452)	NV	Sets the velocity threshold for the controlled stop.
Digital Input (DIN)		
DIN.HCMD1 TO DIN.HCMD4 (pg 455)	NV	A buffer of commands to be used in digital input "command buffer" mode.
DIN.LCMD1 to DIN.LCMD4 (pg 456)	NV	A buffer of commands to be used in digital input "command buffer" mode.
DIN.ROTARY (PG 457)	R/O	Reads the rotary knob value.
DIN.STATES (PG 458)	R/O	Reads the digital input states.
DIN1.FILTER TO DIN7.FILTER (pg 459)	R/W	Filter mode for digital inputs 1 to 7.
DIO9.INV to DIO11.INV (pg 470)	R/W	Inverting the output voltage of the IO, when in the output direction.

Parameter or Command	Type	Description
DIN1.MODE TO DIN24.MODE (pg 461)	NV	Sets the digital input modes.
DIN1.PARAM TO DIN7.PARAM (pg 463)	R/W	Sets a value used as an extra parameter for digital inputs nodes.
DIN1.STATE TO DIN7.STATE (pg 465)	R/O	Reads a specific digital input state.
DIN21.FILTER to DIN32.FILTER (pg 467)	R/W	Filter mode for digital inputs 21 to 32.
DIN21.STATE to DIN32.STATE (pg 468)	R/O	Reads a specific digital input state.
DIO		
DIO9.INV to DIO11.INV (pg 470)	NV	Inverting the output voltage of the IO, when in the output direction.
DIO9.DIR to DIO11.DIR (pg 471)	NV	Changing direction of the IOs from the X9 connector.
Digital Output (DOUT)		
DOUT.CTRL (PG 473)	NV	Sets the source of digital outputs (firmware or fieldbus).
DOUT.RELAYMODE (pg 474)	R/W	Indicates faults relay mode.
DOUT.STATES (PG 475)	R/O	Reads the state of the two digital outputs.
DOUT1.MODE to DOUT19.MODE (pg 476)	NV	Sets the digital output mode.
DOUT1.PARAM AND DOUT2.PARAM (pg 478)	NV	Sets extra parameters for the digital outputs.
DOUT1.STATE AND DOUT2.STATE (pg 480)	R/O	Reads the digital output state.
DOUT1.STATEU AND DOUT2.STATEU (pg 481)	R/W	Sets the state of the digital output node.
DOUT21.STATE to DOUT32.STATE (pg 485)	R/O	Reads the digital output state.
DOUT21.STATEU to DOUT32.STATEU (pg 486)	R/W	Sets the state of the digital output node.
Drive (DRV)		
DRV.ACC (PG 489)	NV	Describes the acceleration ramp for the velocity loop.
DRV.ACTIVE (PG 491)	R/O	Reads the enable status of an axis.
DRV.BLINKDISPLAY (PG 492)	Command	Causes the display to blink for 10 seconds.
DRV.BOOTTIME (pg 493)	R/O	Returns the time when the current session booted up.
DRV.CLRFAULTHIST (PG 494)	Command	Clears the fault history log in the NV.
DRV.CLRFAULTS (PG 495)	Command	Tries to clear all active faults in the drive.
DRV.CMDDELAY (pg 496)	R/W	Issues a delay before next command is executed.
DRV.CMDSOURCE (PG 497)	NV	Sets the command source (service, fieldbus, analog input, gearing, digital, or Bode).
DRV.CRASHDUMP (pg 499)	Command	Retrieves diagnostic information after the drive crashes.
DRV.DBILIMIT (pg 500)	NV	Sets the maximum amplitude of the current for dynamic braking.

Parameter or Command	Type	Description
DRV.DEC (PG 501)	NV	Sets the deceleration value for the velocity loop.
DRV.DIFVAR (pg 503)	R/O	Lists all parameters which differ from their default value.
DRV.DIR (pg 504)	R/W	Changes drive direction.
DRV.DIS (PG 506)	Command	Disables the axis (software).
DRV.DISMODE (pg 507)	NV	Selects among disable immediately or stop and then disable options.
DRV.DISSOURCES (PG 509)	R/O	Returns the possible reason for a drive disable.
DRV.DISSOURCESMASK (pg 510)	R/O	Returns a bitmap of all bits in DRV.DISSOURCES that can be set to one on a given drive.
DRV.DISTO (pg 511)	R/W	Sets the emergency timeout
DRV.EMUEDIR (pg 513)	R/W	Sets the direction of the emulated encoder output (EEO) signal.
DRV.EMUEMODE (pg 514)	R/W	Sets the mode of the emulated encoder output (EEO) connector.
DRV.EMUEMTURN (pg 516)	R/W	Defines the location of the index pulse on the EEO (emulated encoder output) when DRV.EMUEMODE=2.
DRV.EMUEPULSEWIDTH (pg 517)		Sets the encoder output pulse width for modes 6 to 7.
DRV.EMUERES (pg 518)	R/W	Sets the resolution of the EEO (emulated encoder output).
DRV.EMUEZOFFSET (pg 519)	R/W	Sets the location of the EEO (emulated encoder output) index pulse (when DRV.EMUEMODE=1).
DRV.EN (PG 520)	Command	Enables the axis (software).
DRV.ENDEFAULT (pg 521)	R/W	Sets the default state of the software enable.
DRV.FAULTHIST (PG 522)	R/O	Reads the last 10 faults from NV memory.
DRV.FAULTS (PG 524)	R/O	Reads the active faults.
DRV.FAULT1 to DRV.FAULT10 (pg 523)	R/O	Location of fault codes for any active fault conditions.
DRV.HANDWHEEL (pg 525)	R/O	Reads the EEO input value.
DRV.HANDWHEELSRC (pg 526)	NV	Selects the feedback for handwheel operation.
DRV.HELP (PG 527)	R/O	Reads the minimum, maximum, and default values for a specific parameter or command.
DRV.HELPALL (pg 528)	R/O	Retrieves the minimum, maximum, default, and actual values for all available parameters and commands.
DRV.HWENABLE (pg 529)	R/O	Status of the hardware enable.
DRV.HWENDELAY (pg 530)	NV	Delay time between inactive Hardware Enable input and drive disable.
DRV.HWENMODE (pg 531)	R/W	Selects the action that the hardware enable digital input will perform.
DRV.ICONT (PG 532)	R/O	Reads the continuous rated current value.
DRV.INFO (PG 533)	R/O	Reads general information about the drive.
DRV.IPEAK (PG 535)	R/O	Reads the peak rated current value.
DRV.IZERO (pg 536)	R/W	Sets the current that will be used during the DRV.ZERO procedure.

Parameter or Command	Type	Description
DRV.LIST (PG 537)	R/O	Reads the list of available parameters and commands.
DRV.LOGICVOLTS (pg 538)	R/O	Reads the logic voltages.
DRV.NAME (PG 543)	NV	Sets and reads the name of the drive.
DRV.NVCHECK (pg 544)	R/O	NV Parameter Checksum
DRV.NVLIST (PG 545)	R/O	Lists the NV parameters and values from the RAM.
DRV.NVLOAD (pg 546)	W/O	Loads all data from the NV memory of the drive into the RAM parameters.
DRV.NVSAVE (PG 547)	Command	Saves the drive parameters from the RAM to the NV memory.
DRV.ONTIME (pg 548)	R/O	Returns how long the drive has been running since last power up.
DRV.OPMODE (PG 549)	NV	Sets the drive operation mode (current, velocity, or position).
DRV.READFORMAT (PG 551)	R/W	Sets the value returned to either decimal or hexadecimal.
DRV.RSTVAR (PG 552)	Command	Sets default values in the drive without re-booting the drive and without resetting the NV memory.
DRV.RUNTIME (PG 553)	R/O	Returns how long the drive has been running since first acti-vated.
DRV.SETUPREQBITS (pg 554)	R/O	Reads the bitwise set status of parameters that must be set before the drive can be enabled.
DRV.SETUPREQLIST (pg 555)	R/O	Reads the list of parameters that must be set before the drive can be enabled.
DRV.STOP (PG 556)	Command	This command stops all drive motion.
DRV.TEMPERATURES (pg 557)	R/O	Reads the temperature of drive components.
DRV.TIME (pg 558)	R/W	A continuous time counter in the drive.
DRV.TYPE (pg 559)	R/O	Selects the operational fieldbus on CC drive models.
DRV.VER (PG 561)	R/O	Reads the drive version.
DRV.VERIMAGE (PG 562)	R/O	Returns the version data from each image.
DRV.WARNINGS (pg 564)	R/O	Reads the active warnings.
DRV.WARNING1 to DRV.WARNING10 (pg 563)	R/O	Location of fault codes for any active warning conditions.
DRV.ZERO (pg 565)	R/W	Sets the zero mode. The procedure is activated when the drive is enabled.
EtherNet/IP (EIP)		
EIP.CONNECTED (pg 567)	R/O	Returns state of EtherNet/IP connection.
EIP.POSUNIT (pg 568)	R/W	Unit scaling for Position values over EtherNet/IP.
EIP.PROFUNIT (pg 569)	R/W	Unit scaling for Velocity and Acceleration values over Eth-erNet/IP.
Fault (FAULT)		
FAULTx.ACTION (pg 571)	R/W	Gets/Sets the Fault Action for Fault 130, 131, 132, 134, 139, 451, and 702.
Feedback 1 (FB1)		
FB1.BISSBITS (pg 573)	NV	Specifies the number of Biss Sensor (Position) Bits for the BiSS Mode C encoder in use.

Parameter or Command	Type	Description
FB1.DIAG (pg 574)	R/O	Returns feedback diagnostic information.
FB1.ENCRES (PG 575)	NV	Sets the resolution of the motor encoder.
FB1.FAULTS (pg 577)	R/O	Returns a string describing the root cause of Fault 467.
FB1.HALLSTATE (PG 579)	R/O	Reads the Hall switch values (encoder feedback
FB1.HALLSTATEU (pg 580)	R/O	Reads the state of Hall switch U.
FB1.HALLSTATEV (pg 581)	R/O	Reads the state of Hall switch V.
FB1.HALLSTATEW (pg 582)	R/O	Reads the state of Hall switch W.
FB1.IDENTIFIED (PG 583)	R/O	Reads the type of feedback device used by the drive/motor.
FB1.INITSIGNED (pg 584)	NV	Sets initial feedback value as signed or unsigned.
FB1.MECHPOS (PG 585)	R/O	Reads the mechanical position.
FB1.MEMVER (PG 586)	R/O	Returns the memory feedback version.
FB1.OFFSET (pg 587)	NV	Sets position feedback offset.
FB1.ORIGIN (pg 588)	NV	Adds to the initial feedback position.
FB1.P (pg 590)	R/O	Reads position from the primary feedback.
FB1.PDIR (pg 591)	NV	Sets the counting direction for feedback channel 1.
FB1.PFIND (pg 592)	R/W	A procedure that allows the user to find the commutation angle for encoder feedback, which has no halls.
FB1.PFINDCMDU (pg 593)	R/W	Current value used during the phase finding procedure (PFB.PFIND=1)
FB1.PMTBITS (pg 594)	R/O	Returns the number of multiturn bits provided by the feedback device.
FB1.PMTSAVEEN (pg 595)	R/W	Initializes memory for multiturn overflow.
FB1.POFFSET (pg 596)	NV	Sets the offset for primary feedback.
FB1.POLES (PG 597)	R/O	Reads the number of feedback poles.
FB1.PSCALE (pg 598)	R/W	Sets position scaling value for fieldbus transferred position objects.
FB1.PSTBITS (pg 599)	R/O	Returns the number of singleturn bits provided by the feedback device
FB1.PUNIT (pg 600)	NV	Sets the unit for FB1.P.
FB1.RESKTR (pg 601)	NV	Sets the resolver nominal transformation ratio.
FB1.RESREFPHASE (pg 602)	NV	Sets the electrical degrees of phase lag in the resolver.
FB1.SELECT (PG 603)	NV	Sets user entered type or identified type (–1).
FB1.TRACKINGCAL (pg 606)	NV	Controls tracking calibration algorithm.
FB1.USERBYTE0 to FB1.USERBYTE7 (pg 607)	R/W	Reads and writes data stored in two 32 bit words in the Endat feedback device.
FB1.USERDWORD0 to FB1.USERWORD1 (pg 608)	R/W	Reads and writes data stored in two 32 bit words in the Endat feedback device.
FB1.USERWORD1 to FB1.USERWORD3 (pg 609)	R/W	Reads and writes data stored in two 32 bit words in the Endat feedback device.

Parameter or Command	Type	Description
Feedback 2 (FB2)		
FB2.ENCRES (pg 611)	NV	Sets the secondary feedback (FB2) resolution (also defines resolution of virtual encoder in AKD BASIC).
FB2.MODE (pg 612)	R/W	Sets the mode for the second feedback inputs, EEO connector (X9) and high speed opto inputs (pins 9 and 10 on X7).
FB2.P (pg 613)	R/O	Reads position from the secondary feedback.
FB2.DIR (pg 614)	R/W	Sets the counting direction for feedback channel 2.
FB2.POFFSET (pg 615)	NV	Sets the offset for secondary feedback.
FB2.PUNIT (pg 616)	NV	Sets the unit for FB2.P.
FB2.SOURCE (pg 617)	R/W	Sets the source for the second feedback input. Choices are the EEO connectors (X9) which are RS485 inputs, or the X7 connector's high speed opto inputs (pins 9 and 10).
Feedback 3 (FB3)		
FB3.MODE (pg 619)	NV	Selects the type of feedback connected to X9.
FB3.P (pg 620)	RO	Reads position from the tertiary feedback.
FB3.PDIR (pg 621)	NV	Sets the counting direction for feedback channel 3.
FB3.POFFSET (pg 622)	NV	Sets the offset for tertiary feedback.
FB3.PUNIT (pg 623)	NV	Sets the unit for FB3.P.
Fieldbus (FBUS)		
FBUS.PARAM1 TO FBUS.PARAM10 (pg 625)	NV	Set fieldbus specific meanings.
FBUS.PLLSTATE (pg 628)	R/O	Returns the status of the PLL
FBUS.PLLTHRESH (pg 629)	NV	Sets number of successful synchronized cycles needed to lock the PLL.
FBUS.PROTECTION (pg 630)	R/W	Controls which parameters are blocked from being accessed through telnet while a fieldbus is operational.
FBUS.SAMPLEPERIOD (pg 633)	NV	Sets fieldbus sample period.
FBUS.STATE (pg 634)	R/O	Reads the state of the fieldbus.
FBUS.SYNCACT (pg 635)	R/O	Reads actual distance from the desired sync distance.
FBUS.SYNCDIST (pg 636)	NV	Sets time target for synchronization.
FBUS.SYNCWND (pg 637)	NV	Sets symmetrically arranged window around the desired sync distance.
FBUS.TYPE (pg 638)	R/O	Shows the active fieldbus type.
Gearing (GEAR)		
GEAR.ACCMAX (pg 640)	R/W	Sets the maximum allowed acceleration value; active in opmode 2 (position) only.
GEAR.DECMAX (pg 641)	R/W	Sets the maximum allowed deceleration value; active in opmode 2 (position) only.
GEAR.IN (pg 643)	R/W	Sets the denominator of the electronic gearing ratio; active in opmode 2 (position) only.
GEAR.MODE (pg 644)	R/W	Selects electronic gearing mode; active in opmode 2 (position) only.
GEAR.MOVE (pg 646)	Command	Starts the electronic gearing; active in opmode 2 (position) only.
GEAR.OUT (pg 647)	R/W	Sets the numerator of the electronic gearing ratio; active in opmode 2 (position) only.

Parameter or Command	Type	Description
GEAR.VMAX (pg 648)	R/W	Reads the maximum allowed velocity value; active in opmode 2 (position) only.
Homing (HOME)		
HOME.ACC (pg 663)	R/W	Sets homing acceleration; active in opmode 2 (position) only.
HOME.AUTOMOVE (pg 665)	R/W	Sets homing auto move flag.
HOME.DEC (pg 666)	R/W	Sets homing deceleration; active in opmode 2 (position) only.
HOME.DIR (pg 668)	NV	Sets homing direction; active in opmode 2 (position) only.
HOME.DIST (pg 669)	R/W	Sets homing distance; active in opmode 2 (position) only.
HOME.FEEDRATE (pg 670)	R/W	Sets homing velocity factor; active in opmode 2 (position) only.
HOME.IPEAK (pg 671)	R/W	Sets the current limit during homing procedure to a mechanical stop; active in opmode 2 (position) only.
HOME.MODE (pg 672)	R/W	Selects the homing mode; active in opmode 2 (position) only.
HOME.MOVE (pg 674)	Command	Starts a homing procedure; active in opmode 2 (position) only.
HOME.P (pg 675)	R/W	Sets home position; active in opmode 2 (position) only.
HOME.PERRTHRESH (pg 676)	R/W	Sets the position lag threshold; active in opmode 2 (position) only.
HOME.REQUIRE (pg 677)	NV	Defines if the axis must be homed before a motion task can be executed.
HOME.SET (pg 678)	Command	Immediately sets the home position; active in opmode 2 (position) only.
HOME.V (pg 679)	R/W	Sets homing velocity; active in opmode 2 (position) only.
Hardware Limit Switch (HWLS)		
HWLS.NEGSTATE (pg 682)	R/O	Reads the status of the negative hardware limit switch.
HWLS.POSSTATE (pg 683)	R/O	Reads the status of the positive hardware limit switch.
Current Loop (IL)		
IL.BUSFF (pg 686)	R/O	Displays the current feedforward value injected by the fieldbus.
IL.CMD (PG 687)	R/O	Reads the value of the q-component current command.
IL.CMDACC (pg 688)	R/O	Returns the commanded acceleration from the trajectory generator.
IL.CMDU (PG 689)	R/W	Sets the user current command.
IL.DIFOLD (PG 690)	R/O	Reads the drive foldback current limit.
IL.FB (PG 691)	R/O	Reads the actual value of the d-component current.
IL.FBSOURCE (pg 692)	R/W	Sets the feedback source for the current loop. Only applies when MOTOR.TYPE = 4.
IL.FF (pg 693)	R/O	Displays the current loop overall feedforward value.
IL.FOLDFTHRESH (PG 694)	NV	Reads the foldback fault level.
IL.FOLDFTHRESHU (pg 695)	NV	Sets the user value for the foldback fault level.
IL.FOLDWTHRESH (PG 696)	NV	Sets the foldback warning level.

Parameter or Command	Type	Description
IL.IFOLD (pg 698)	R/O	Reads the overall foldback current limit.
IL.IUFB (PG 699)	R/O	Reads the sigma-delta measured current in the u-winding of the motor.
IL.KACCFF (pg 701)	R/W	Sets current loop acceleration feedforward gain value
IL.KBUSFF (pg 702)	R/W	Current loops fieldbus injected feed-forward gain
IL.KP (PG 703)	NV	Sets the proportional gain of the q-component of the PI regulator.
IL.KPDRATIO (PG 704)	NV	Sets the proportional gain of the d-component current PI-regulator as a percentage of IL.KP
IL.KPLOOKUPINDEX (pg 705)	R/W	Sets the index into the Current Loop Gain Scheduling Table.
IL.KPLOOKUPVALUE (pg 706)	R/W	Sets the value of the current loop gain scheduling index.
IL.KPLOOKUPVALUES (pg 707)	R/W	Gets the Current Loop Gain Scheduling Table.
IL.KVFF (pg 708)	R/W	Current loop velocity feed-forward gain.
IL.LIMITN (PG 709)	NV	Sets the negative user (application-specific) current limit.
IL.LIMITP (PG 710)	NV	Sets the positive user (application-specific) current limit.
IL.MFOLDD (PG 711)	NV	Sets the motor foldback maximum time at motor peak current.
IL.MFOLDR (PG 712)	R/O	Sets the motor foldback recovery time.
IL.MFOLDT (PG 713)	NV	Sets the motor foldback time constant of the exponential current drop (foldback).
IL.MI2T (pg 714)	R/O	Motor I2t load.
IL.MI2TWTHRESH (pg 715)	NV	Motor I2t load warning threshold.
IL.MIFOLD (PG 716)	R/O	Sets the motor foldback current limit.
IL.MIMODE (pg 717)	NV	Motor protection mode.
IL.OFFSET (pg 718)	RW	A constant current command added to compensate for gravity.
IL.VCMD (PG 719)	R/O	Sets the output of the q-component PI regulator.
IL.VUFB (PG 720)	R/O	Reads the measured voltage on the u-winding of the motor.
IL.VVFB (PG 721)	R/O	Reads the measured voltage on the v-winding of the motor.
IP (Internet Protocol) Parameters		
IP.ADDRESS (pg 723)	NV	Gets/Sets the IP address of the drive.
IP.GATEWAY (pg 724)	NV	Gets/Sets the gateway IP of the drive.
IP.MODE (pg 725)	NV	Sets method of acquiring IP Address.
IP.RESET (pg 727)	Command	Implements new IP settings..
IP.SUBNET (pg 729)	NV	Gets/Sets the IP Subnet mask of the drive.
LOAD Parameters		
LOAD.INERTIA (pg 731)	NV	Sets the load inertia.
MODBUS Parameters		
MODBUS.PIN (pg 736)	R/W	Gets / Sets the Modbus User Units Input parameter.
MODBUS.CLRERRORS (pg 733)	Command	Clears all errors stored in MODBUS.ERRORS.
MODBUS.ERRORMODE (pg 734)	R/W	Enable/disable Modbus error response messages.

Parameter or Command	Type	Description
MODBUS.ERRORS (pg 735)	R/O	Returns a list of up to 125 Modbus errors.
MODBUS.POUT (pg 737)	R/W	Gets / Sets the Modbus User Units Output parameter.
MODBUS.PSCALE (pg 738)	R/W	Gets/Sets the Feedback Resolution (per rev) over Modbus.
MODBUS.SCALING (pg 739)	NV	Selects the scaling mode for Modbus values.
MODBUS.UNITLABEL (pg 740)	R/W	Labels the scaled resolution of a single motor turn.

Parameter or Command	Type	Description
Motor Parameters		
MOTOR.AUTOSET (pg 742)	NV	Determines which drive parameters are calculated automatically.
MOTOR.BRAKE (PG 744)	NV	Sets the presence or absence of a motor brake.
MOTOR.BRAKEIMM (pg 745)	NV	Brake Immediately: in the case of a drive disable, apply the brake in all situations.
MOTOR.BRAKERLS (pg 746)	Command	Allows a user to release or apply the motor brake.
MOTOR.BRAKESTATE (pg 747)	R/O	Reads the actual status of the motor brake.
MOTOR.CTF0 (pg 748)	NV	Sets the thermal constant of the motor coil.
MOTOR.ICONT (PG 749)	NV	Sets the motor continuous current.
MOTOR.IDDATAVALID (pg 750)	R/O	Reports the status of the motor memory.
MOTOR.IMID (pg 751)	R/W	The direct-axis current set point used for induction machine closed-loop control.
MOTOR.IMTR (pg 752)	R/W	Rotor time constant.
MOTOR.INERTIA (PG 754)	NV	Sets the motor inertia.
MOTOR.IPEAK (PG 755)	NV	Sets the motor peak current.
MOTOR.KE (pg 756)		Sets the motor back EMF constant.
MOTOR.KT (PG 757)	NV	Sets the torque constant of the motor.
MOTOR.LQLL (PG 758)	NV	Sets the line-to-line motor Lq.
MOTOR.NAME (PG 759)	NV	Sets the motor name.
MOTOR.PHASE (PG 760)	NV	Sets the motor phase.
MOTOR.PITCH (PG 761)	NV	Sets the motor pitch.
MOTOR.POLES (PG 762)	NV	Sets the number of motor poles.
MOTOR.R (PG 763)	NV	Sets the stator winding resistance phase-phase in ohms.
MOTOR.RTYPE (pg 764)	NV	Defines the type of thermal resistor inside the motor.
MOTOR.TBRAKEAPP (PG 765)	NV	The delay time used for applying the motor brake.
MOTOR.TBRAKERLS (PG 766)	NV	The delay time used for releasing the motor brake.
MOTOR.TBRAKETO (pg 767)	NV	Brake apply timeout for vertical axis.
MOTOR.TEMP (pg 768)	R/O	Reads the motor temperature represented as the resistance of the motor PTC or NTC.

Parameter or Command	Type	Description
MOTOR.TEMPFAULT (pg 769)	NV	Sets the motor temperature fault level.
MOTOR.TEMPWARN (pg 770)	NV	Sets the motor temperature warning level.
MOTOR.TYPE (PG 771)	NV	Sets the motor type.
MOTOR.VMAX (PG 773)	NV	Sets the maximum motor speed.
MOTOR.VOLTMAX (PG 774)	NV	Sets the motor maximum voltage.
MOTOR.VOLTMIN (pg 775)	NV	Sets the minimum voltage for V/f control.
MOTOR.VOLTRATED (pg 776)	NV	Sets the motor rated voltage.
MOTOR.VRATED (pg 777)	NV	Sets the motor rated velocity (not maximum velocity).
Motion Task (MT)		
MT.ACC (PG 780)	R/W	Specifies motion task acceleration; active in opmode 2 (position) only.
MT.CLEAR (PG 782)	Command	Clears motion tasks from the drive; active in opmode 2 (position) only.
MT.CNTL (PG 783)	R/W	Sets motion task control word; active in opmode 2 (position) only.
MT.CONTINUE (PG 786)	Command	Continues a stopped motion task; active in opmode 2 (position) only.
MT.DEC (PG 787)	R/W	Sets motion task deceleration; active in opmode 2 (position) only.
MT.EMERGMT (PG 789)	R/W	Selects a motion task to be triggered after an emergency stop procedure; active in opmode 2 (position) only.
MT.HOMEREQUIRE (pg 790)	NV	Removed in 01-04-00-000.
MT.LIST (pg 791)	Command	Lists all initialized motion tasks in the drive; active in opmode 2 (position) only.
MT.LOAD (PG 792)	Command	Reads/loads a motion task number from the drive; active in opmode 2 (position) only.
MT.MOVE (PG 793)	Command	Starts a motion task; active in opmode 2 (position) only.
MT.MTNEXT (PG 794)	R/W	Specifies following motion task number; active in opmode 2 (position) only.
MT.NUM (pg 795)	R/W	Sets the motion task number; active in opmode 2 (position) only.
MT.P (PG 796)	R/W	Sets the motion task position; active in opmode 2 (position) only.
MT.PARAMS (PG 797)	Command	Shows a motion task; active in opmode 2 (position) only.
MT.SET (PG 798)	Command	Sets the motion task in the drive; active in opmode 2 (position) only.
MT.TDWNLDS (Password Protected)	W/O	Motion profile table download status
MT.TDWNLDV (Password Protected)	W/O	Sets motion profile table download value.
MT.TNEXT (PG 799)	R/W	Specifies following motion task time; active in opmode 2 (position) only.

Parameter or Command	Type	Description
MT.NUM (PG 795)	R/W	Sets the motion task customer table number; active in opmode 2 (position) only.
MT.TNVSAVE (pg 801)	Command	Saves the motion profile tables to the nonvolatile memory.
MT.TPOSWND (pg 802)	R/W	Sets the motion task target position window; active in opmode 2 (position) only.
MT.TRETRIEVE (Password Protected)	R/O	Retrieves motion profile table.
MT.TRETRIEVEDATA (Password Protected)	R/W	Retrieves motion profile table data.
MT.TRETRIEVESIZE (Password Protected)	R/W	Retrieves motion profile table size.
MT.TSIZE (Password Protected)	R/O	Gets the motion profile table size.
MT.TVELWND (pg 803)	R/W	Sets the motion task target velocity window; active in opmode 2 (position) only.
MT.T0NAME TO MT.T7NAME (Password Protected)	R/W	Names a motion profile table.
MT.V (PG 805)	R/W	Sets the motion task velocity; active in opmode 2 (position) only.
MT.VCMD (pg 807)	R/O	Reads the derivative of PL.CMD; active in opmode 2 (position) only.
Position Loop (PL)		
PL.CMD (PG 809)	NV	Reads the position command directly from the entry to the position loop.
PL.ERR (PG 810)	NV	Reads the position error present when the drive is controlling the position loop.
PL.ERRFTHRESH (pg 811)	NV	Sets the maximum position error.
PL.ERRMODE (pg 813)	R/W	Sets the type of following error warning and fault usage.
PL.ERRWTHRESH (pg 815)	NV	Sets the position error warning level.
PL.FB (PG 817)	R/O	Reads the position feedback value.
PL.FBSOURCE (pg 818)	NV	Sets the feedback source for the position loop.
PL.FILTERTIME (pg 819)	R/W	Sets the position command filter period.
PL.INTINMAX (PG 820)	NV	Limits the input of the position loop integrator by setting the input saturation.
PL.INTOUTMAX (PG 822)	NV	Limits the output of the position loop integrator by setting the output saturation.
PL.KI (PG 824)	NV	Sets the integral gain of the position loop.
PL.KP (PG 825)	NV	Sets the proportional gain of the position regulator PID loop.
PL.MODP1 (pg 826)	R/W	Sets modulo range parameter.
PL.MODP2 (pg 827)	R/W	Sets the beginning or end modulo range parameter.
PL.MODPDIR (pg 828)	R/W	Sets the direction for absolute motion tasks.
PL.MODPEN (pg 829)	R/W	Enables the modulo position.
Programmable Limit Switch (PLS)		
PLS.EN (pg 831)	R/W	Enables programmable limit switch (PLS).

Parameter or Command	Type	Description
PLS.MODE (pg 832)	NV	Selects programmable limit switch mode.
PLS.P1 TO PLS.P8 (pg 833)	NV	Sets the trigger point for programmable limit switches.
PLS.RESET (pg 835)	W/O	Resets programmable limit switch.
PLS.STATE (pg 836)	R/O	Reads the programmable limit switch state.
PLS.T1 TO PLS.T8 (pg 837)	R/W	Sets programmable limit switch time.
PLS.UNITS (pg 839)	R/W	Sets programmable limit switch (PLS) units.
PLS.WIDTH1 TO PLS.WIDTH8 (pg 841)	R/W	Programmable Limit Switch Width
Recorder (REC)		
REC.ACTIVE (PG 844)	R/O	Indicates if data recording is in progress (active).
REC.CH1 to REC.CH6 (pg 845)	R/W	Sets recording channels 1 to 6.
REC.DONE (PG 846)	R/O	Checks whether or not the recorder has finished recording.
REC.GAP (PG 847)	R/W	Specifies the gap between consecutive samples.
REC.NUMPOINTS (PG 848)	R/W	Sets the number of points to record.
REC.OFF (PG 849)	R/W	Turns the recorder OFF.
REC.RECPRMLIST (pg 850)	R/O	Reads the list of recordable parameters.
REC.RETRIEVE (PG 851)	R/O	Transfers all the recorded data to the communication channel.
REC.RETRIEVEDATA (pg 852)	R/W	Retrieves the recorded data without the header.
REC.RETRIEVEFRMT (pg 854)	R/W	Sets the format for recorded data output.
REC.RETRIEVEHDR (pg 855)	R/O	Retrieves the recorded header without the data.
REC.RETRIEVESIZE (pg 856)	R/W	Sets the number of samples that REC.RETRIEVEDATA returns.
REC.STOPTYPE (pg 857)	R/W	Sets the recorder stop type.
REC.TRIG (PG 858)	Command	Triggers the recorder.
REC.TRIGPARAM (pg 859)	R/W	Sets the parameter that triggers the recorder.
REC.TRIGPOS (pg 860)	R/W	Sets the trigger position in the recording buffer.
REC.TRIGPRMLIST (pg 862)	R/O	Reads the list of possible trigger parameters.
REC.TRIGSLOPE (PG 863)	R/W	Sets the trigger slope.
REC.TRIGTYPE (PG 864)	R/W	Sets the trigger type.
REC.TRIGVAL (PG 865)	R/W	Sets the trigger value.
Regen Resistor (REGEN)		
REGEN.POWER (PG 867)	R/O	Reads regen resistor's calculated power.
REGEN.POWERFILTERED (pg 868)	R/O	Returns a filtered version of REGEN.POWER .
REGEN.REXT (PG 869)	N/V	Sets the external, user-defined regen resistor resistance.
REGEN.TEXT (pg 870)	R/W	Sets the external regen resistor thermal protection time constant.
REGEN.TYPE (PG 872)	N/V	Sets the regen resistor type.

Parameter or Command	Type	Description
REGEN.WATTEXT (PG 873)	R/W	Sets the regen resistor's power fault level for an external regen resistor.
SD card (SD)		
SD.LOAD (pg 875)	Command	Loads the drive state (BASIC program and NV parameters) from the SD card to the AKD (AKDs equipped with IO option card only).
SD.SAVE (pg 876)	Command	Saves the drive state (BASIC program and NV parameters) to the SD card (AKDs equipped with IO option card only).
SD.STATUS (pg 877)	R/O	Reads the status of the SD card.
Service Motion (SM)		
SM.I1 (pg 879)	R/W	Sets service motion current 1; active in opmode 0 (torque) only.
SM.I2 (pg 880)	R/W	Sets service motion current 2; active in opmode 0 (torque) only.
SM.MODE (pg 881)	R/W	Sets the service motion mode.
SM.MOVE (pg 884)	Command	Starts the service motion.
SM.T1 (pg 885)	R/W	Sets the service motion time 1.
SM.T2 (pg 886)	R/W	Sets the service motion time 2.
SM.V1 (pg 887)	R/W	Sets service motion velocity 1; active in opmode 1 (velocity) and 2 (position).
SM.V2 (pg 889)	R/W	Sets service motion velocity 2; active in opmode 1 (velocity) and 2 (position).
Safe Torque Off (STO)		
STO.STATE (pg 892)	R/O	Returns the status of the safe torque off.
Software Limit Switch (SWLS)		
SWLS.EN (pg 894)	NV	Enables and disables software travel limit switches.
SWLS.LIMIT0 (pg 895)	NV	Sets the position of the software travel limit switch 0.
SWLS.LIMIT1 (pg 896)	NV	Sets the position of the software travel limit switch 0.
SWLS.STATE (pg 897)	R/O	Reads the actual status of software limit switches.
Temperature (TEMP)		
TEMP.CONTROL (pg 899)	R/O	Reads the temperature of the control board temperature sensor.
TEMP.POWER1 to TEMP.POWER3 (pg 900)	R/O	Reads the temperature of the power board(s) temperature sensor(s).
Units (UNIT)		
UNIT.ACCLINEAR (PG 902)	NV	Sets the linear acceleration/deceleration units.
UNIT.ACCROTARY (PG 903)	NV	Sets the rotary acceleration/deceleration units.
UNIT.LABEL (pg 904)	NV	Sets user-defined name for user-defined position units.
UNIT.PIN (PG 905)	NV	Sets gear IN for the unit conversion.
UNIT.PLINEAR (PG 906)	NV	Sets the linear position units.
UNIT.POUT (PG 907)	NV	Sets gear out for the unit conversion.
UNIT.PROTARY (PG 908)	NV	Sets the position units when the motor type (MOTOR.TYPE (pg 769)) is rotary.
UNIT.VLINEAR (PG 909)	NV	Sets the linear velocity units.

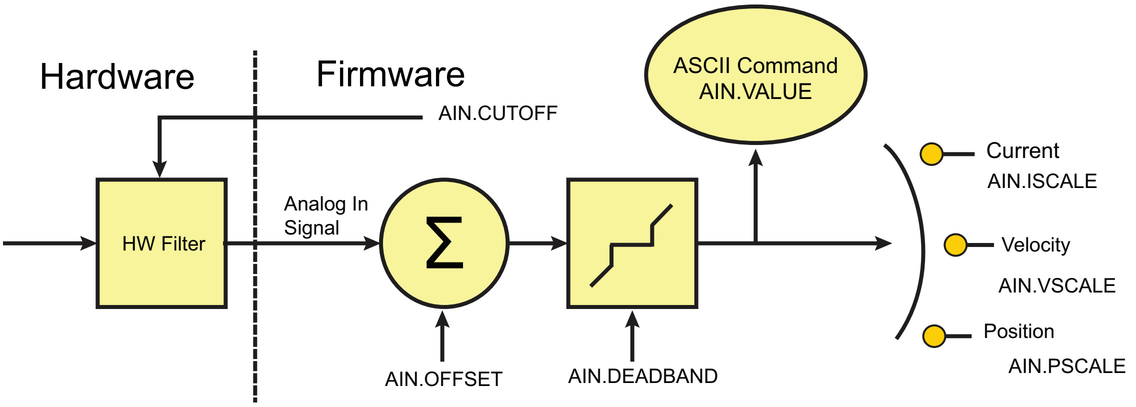
Parameter or Command	Type	Description
UNIT.VROTARY (PG 910)	NV	Sets the velocity units when the motor type (MOTOR.TYPE (pg 769)) is rotary.
Bus Volatage (VBUS)		
VBUS.HALFVOLT (pg 912)	NV	Changing voltage thresholds for HV and MV Drives
VBUS.OVFTHRESH (pg 913)	R/O	Reads the over voltage fault level.
VBUS.OVWTHRESH (pg 914)	N/V	Sets voltage level for over voltage warning.
VBUS.RMSLIMIT (pg 915)	R/O	Reads the limit for the bus capacitors load.
VBUS.UVFTHRESH (pg 916)	R/O	Sets the under voltage fault level.
VBUS.UVMODE (pg 917)	NV	Indicates undervoltage (UV) mode.
VBUS.UVWTHRESH (pg 918)	NV	Sets voltage level for undervoltage warning.
VBUS.VALUE (pg 919)	R/O	Reads DC bus voltage.
Velocity Loop (VL)		
VL.ARPF1 TO VL.ARPF4 (pg 921)	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 (pg 923)	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.A-RTYPE4 (pg 925)	NV	Indicates the method used to calculate BiQuad coefficients; active in opmodes 1 (velocity) and 2 (position) only.
VL.ARZF1 TO VL.ARZF4 (pg 926)	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 (pg 928)	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 (pg 930)	R/O	Displays the velocity loop feedforward value injected by the field-bus; active in opmodes 1 (velocity) and 2 (position) only.
VL.CMD (PG 931)	R/O	Reads the actual velocity command; active in opmodes 1 (velocity) and 2 (position) only.
VL.CMDU (PG 932)	R/W	Sets the user velocity command; active in opmodes 1 (velocity) and 2 (position) only.
VL.ERR (PG 933)	R/O	Sets the velocity error; active in opmodes 1 (velocity) and 2 (position) only.
VL.FB (PG 934)	R/O	Reads the velocity feedback; active in opmodes 1 (velocity) and 2 (position) only.
VL.FBFILTER (pg 935)	R/O	Filters VL.FB (pg 932) value; active in opmodes 1 (velocity) and 2 (position) only.
VL.FBSOURCE (pg 936)	NV	Sets feedback source for the velocity loop; active in opmodes 1 (velocity) and 2 (position) only.
VL.FBUNFILTERED (pg 937)	R/O	Reads the velocity feedback.
VL.FF (pg 938)	R/O	Displays the velocity loop overall feedforward value; active in opmodes 1 (velocity) and 2 (position) only.
VL.GENMODE (PG 939)	NV	Selects mode of velocity generation (Observer, d/dt); active in opmodes 1 (velocity) and 2 (position) only.

Parameter or Command	Type	Description
VL.KBUSFF (pg 940)	R/W	Sets the velocity loop acceleration feedforward gain value; active in opmodes 1 (velocity) and 2 (position) only.
VL.KI (pg 941)	NV	Sets the velocity loop integral gain for the PI controller; active in opmodes 1 (velocity) and 2 (position) only.
VL.KP (pg 944)	NV	Sets velocity loop proportional gain for the PI controller; active in opmodes 1 (velocity) and 2 (position) only.
VL.KVFF (pg 946)	R/W	Sets the velocity loop velocity feedforward gain value; active in opmodes 1 (velocity) and 2 (position) only.
VL.LIMITN (PG 947)	NV	Sets the velocity lower limit; active in opmodes 1 (velocity) and 2 (position) only.
VL.LIMITP (PG 949)	NV	Sets the velocity high limit; active in opmodes 1 (velocity) and 2 (position) only.
VL.LMJR (pg 951)	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 (pg 952)	R/O	Reads the observer velocity signal; active in opmodes 1 (velocity) and 2 (position) only.
VL.OBSBW (pg 953)	NV	Sets the bandwidth of the observer in Hz.
VL.OBSMODE (pg 954)	NV	Sets the observer operating mode.
VL.THRESH (PG 955)	NV	Sets the over speed fault value; active in opmodes 1 (velocity) and 2 (position) only.
Wake and Shake (WS)		
WS.ARM (pg 958)	Command	Sets wake and shake to start at the next drive enable.
WS.CHECKMODE (pg 959)	R/W	Select the type of commutation check to execute after Wake and Shake finds a new commutation angle.
WS.CHECKT (pg 960)	R/W	Sets the amount of time a communication error must be present before an error is thrown.
WS.CHECKV (pg 961)	R/W	This parameter sets the velocity threshold which has to be exceeded to activate commutation monitoring.
WS.DISARM (pg 962)	Command	Cancels ARM requests and resets wake and shake to the IDLE state.
WS.DISTMAX (pg 963)	R/W	Sets maximum movement allowed for wake and shake.
WS.DISTMIN (pg 964)	R/W	Sets the minimum movement required for wake and shake.
WS.FREQ (pg 965)	R/W	Sets the sine frequency of excitation for WS.MODE 2.
WS.IMAX (pg 966)	R/W	Sets maximum current used for wake and shake.
WS.MODE (pg 967)	R/W	Sets the method used for wake and shake.
WS.NUMLOOPS (pg 968)	R/W	Sets the number of repetitions for wake and shake.
WS.STATE (pg 969)	R/O	Reads wake and shake status.
WS.T (pg 970)	R/W	Sets wake and shake current-vector appliance time.
WS.TDELAY1 (pg 971)	NV	Delay for wake and shake timing.
WS.TDELAY2 (pg 972)	NV	Sets the delay for wake and shake timing.
WS.TDELAY3 (pg 973)	NV	Sets the delay for wake and shake between loops in mode 0.
WS.TIRAMP (pg 974)	R/W	Sets the ramp time for the ramp up current in Wake & Shake mode 1.
WS.TSTANDSTILL (pg 975)	R/W	Sets the calming time of the motor for Wake & Shake mode 1.
WS.VTHRESH (pg 976)	NV	Defines the maximum allowed velocity for Wake & Shake.

This page intentionally left blank.

25.2 AIN Parameters

This section describes the analog input (AIN) parameters. AIN parameters function as shown in the block diagram below:



25.2.1 AIN.CUTOFF

General Information	
Type	NV Parameter
Description	Sets the analog input low-pass filter cutoff frequency.
Units	Hz
Range	0 to 10,000 Hz
Default Value	5,000 Hz
Data Type	Float
See Also	AIN Parameters (pg 364)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	0	No	32 bit	No	M_01-03-00-000

Description

AIN.CUTOFF sets the break frequency in Hz for two cascaded single-pole low-pass filters on the hardware command input. Since the two poles are cascaded at the same frequency, the -3 dB frequency is 0.64*AIN.CUTOFF in hertz and the 10% to 90% step response rise time is 0.53/AIN.CUTOFF in seconds.

Suggested operating values are as follows:

- Analog torque opmode: 5 kHz
- Analog velocity opmode: 2.5 kHz
- General purpose analog input high resolution: 500 Hz

Related Topics

Analog Input (pg 115)

25.2.2 AIN.DEADBAND

General Information	
Type	NV Parameter
Description	Sets the analog input signal dead-band.
Units	V
Range	0 to 12.5 V
Default Value	0 V
Data Type	Float
Start Version	M_01-00-00-000

Variant Support

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

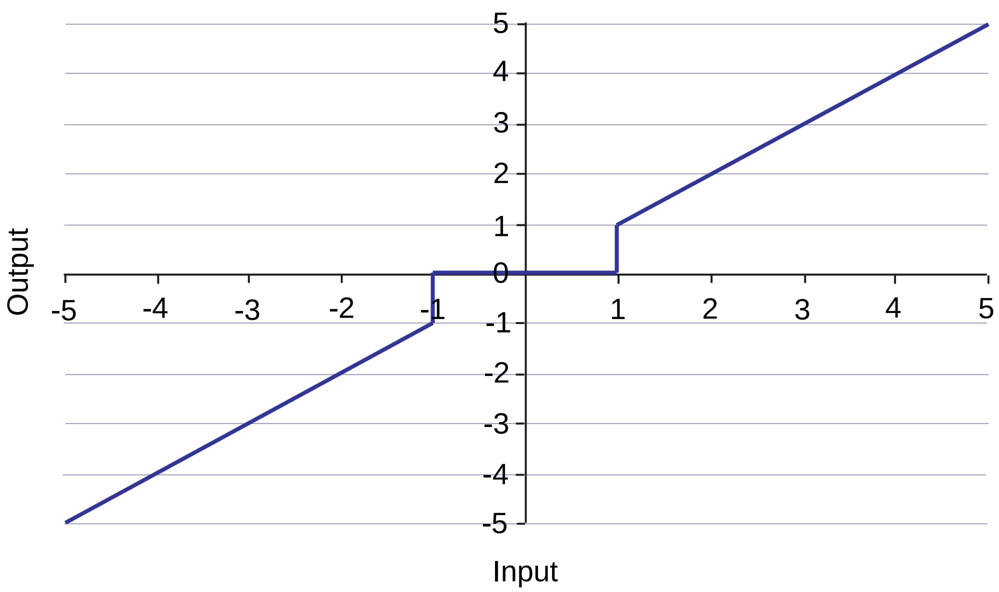
Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	2	No	16 bit	No	M_01-03-00-000

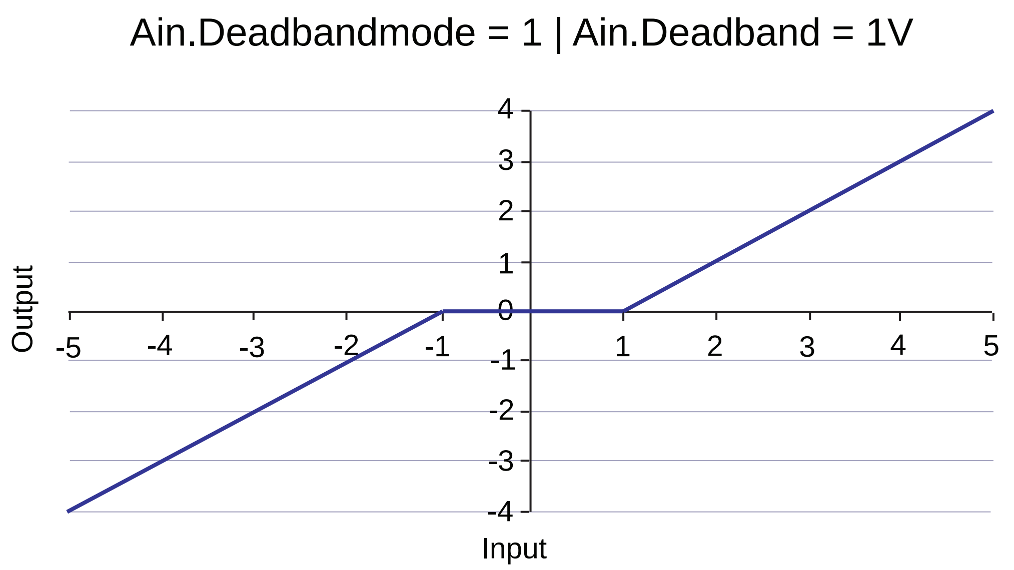
Description

AIN.DEADBAND sets the deadband of the analog input signal. When AIN.DEADBANDMODE is set to 0, and the value of the analog input is less than the value of AIN.DEADBAND, the analog command will be 0. When the analog input is greater or equal to the AIN.DEADBAND, then the analog command will be generated using the scaling specified.

When AIN.DEADBANDMODE is set to 1, the analog command is 0 if the input is less than the deadband value. When the input is greater than the deadband, the output is equal to (Input - Deadband) * Scaling. Below are illustrations of this behavior.

Ain.Deadbandmode = 0 | Ain.Deadband = 1V





Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

AIN Parameters (pg 364)

25.2.3 AIN.DEADBANDMODE

General Information	
Type	NV Parameter
Description	Sets the analog input deadband mode.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Integer
See Also	AIN.DEADBAND (pg 366)
Start Version	M_01-03-06-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	

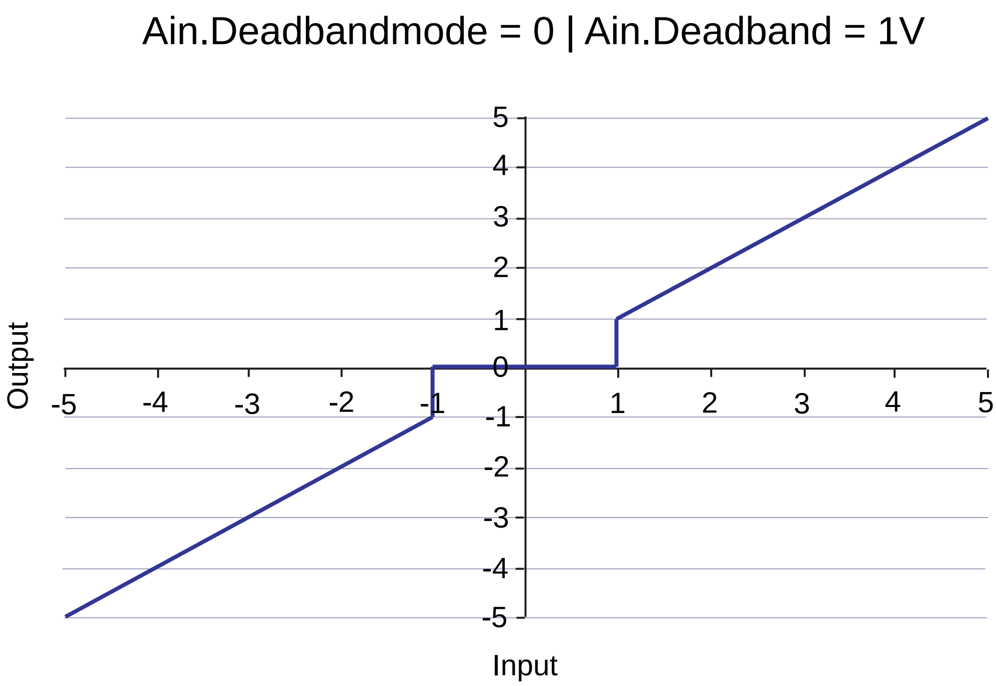
Fieldbus Information

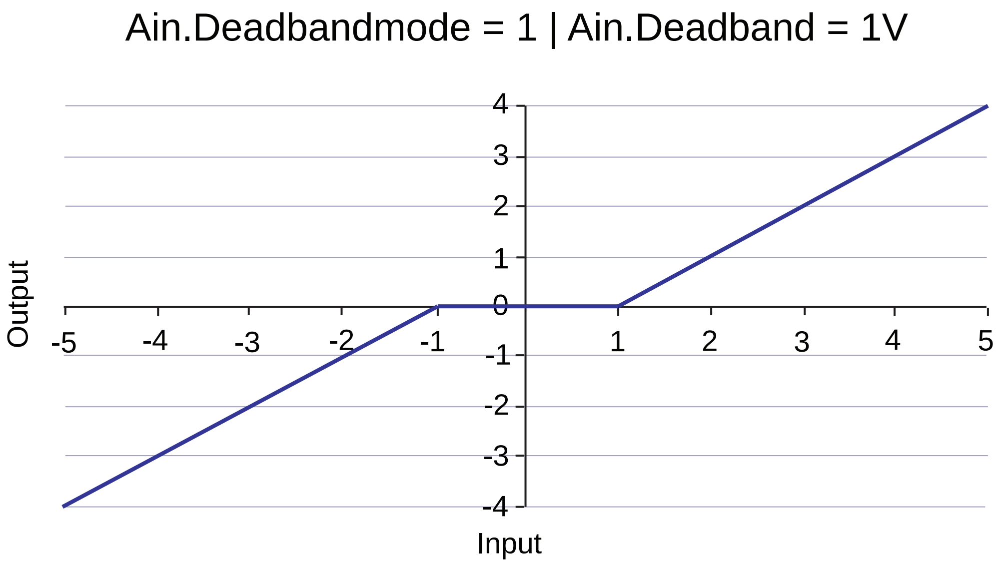
Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1186	No	16 bit	No

Description

When AIN.DEADBANDMODE is set to 0, and the value of the analog input is less than the value of AIN.DEADBAND, the analog command will be 0. When the analog input is greater or equal to the AIN.DEADBAND, then the analog command will be generated using the scaling specified.

When AIN.DEADBANDMODE is set to 1, the analog command is 0 if the input is less than the deadband value. When the input is greater than the deadband, the output is equal to (Input - Deadband) * Scaling. Below are illustrations of this behavior.





Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309)(for the drive controller environment).

25.2.4 AIN.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 A/V
Data Type	Float
See Also	AIN Parameters (pg 364)
Start Version	M_01-01-01-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	4	No	32 bit	No	M_01-03-00-000

Description

AIN.ISCALE sets the analog current scale factor that scales the analog input (AIN.VALUE) for DRV.OPMODE (pg 549) = 1 (analog torque mode).

The value entered is the motor current per 10 V of analog input. This value may be either higher or lower than 100%, but the actual analog input will be limited by the application current limit (IL.LIMITN (pg 709) and IL.LIMITP (pg 710)).

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

25.2.5 AIN.MODE

General Information	
Type	NV Parameter
Description	Analog input mode
Units	N/A
Range	0 to 2
Default Value	1
Data Type	Integer
See Also	AIN Parameters (pg 364)
Start Version	M_01-04-09-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1188	No	8 bit	No

AKD SynqNet	
Range	0

AKD BASIC	
Range	0 to 1

Description

The parameter AIN.MODE is used to assign a functionality to the voltage measured on the analog input pin.

0 – The analog input value is not used by any function.

1 – This mode only works when DRV.CMDSOURCE is set to 3 (analog). The measured voltage will be scaled with:

- AIN.ISCALE if DRV.OPMODE has been set to 0 (torque mode)
- AIN.VSCALE if DRV.OPMODE has been set to 1 (velocity mode)
- AIN.PSCALE if DRV.OPMODE has been set to 2 (position mode)

Afterwards, the value will be forwarded as a command value to the control-loops.

2 – This mode is used for generating a target velocity of a motion task. This mode works when DRV.OPMODE is set to 2 (position) and DRV.CMDSOURCE is set to 0 (service). The measured voltage will be scaled with AIN.VSCALE.

Related Topics

MT.CNTL (pg 783)

DRV.OPMODE (pg 549)

25.2.6 AIN.OFFSET

General Information	
Type	NV Parameter
Description	Sets the analog input offset.
Units	V
Range	–10 to +10 V
Default Value	0 V
Data Type	Float
See Also	AIN Parameters (pg 364), AIN.ZERO (pg 378)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	6	No	16 bit	Yes	M_01-03-00-000

Description

AIN.OFFSET sets the analog offset, which is added to the analog input command to the drive. This value compensates for the analog input signal (AIN.VALUE (pg 375)) offset or drift.

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

25.2.7 AIN.PSCALE

General Information	
Type	NV Parameter
Description	Sets the analog position scale factor.
Units	Depends on UNIT.PROTARY (pg 908) or UNIT.PLINEAR (pg 906) Rotary: counts/V,rad/V, deg/V, (Custom Units (pg 96))/V, 16-bit counts/V Linear: counts/V, mm/V, µm/V, (Custom Units (pg 96))/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 0.06 to 179.0 deg/V 0 to 10,737,418.240 (PIN/POUT)/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 µm/V 0 to 10,737,418.240 (PIN/POUT)/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 (PIN/POUT)/V 0 16-bit counts/V Linear: 1 count/V 0 rad/V 0 deg/V 0 (PIN/POUT)/V 0 16-bit counts/V
Data Type	Float
Start Version	M_01-01-01-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3472h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	8	Yes	64 bit	Yes	M_01-03-00-000

Description

AIN.PSCALE is an analog position scale factor that scales the analog input (AIN.VALUE (pg 375)) for DRV.OPMODE (pg 549) = 2 , DRV.CMDSOURCE (pg 497) = 3 (analog position mode).

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

25.2.8 AIN.VALUE

General Information	
Type	R/O Parameter
Description	Reads the value of the analog input signal.
Units	V
Range	-12.5 to +12.5 V
Default Value	N/A
Data Type	Float
See Also	AIN.OFFSET (pg 372), AIN.ZERO (pg 378), AIN Parameters (pg 364)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3470h/4 3509h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	12	No	16 bit	No	M_01-03-00-000

Description

AIN.VALUE reads the analog input value after the value is filtered (as shown in the Analog Input Block Diagram).

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

25.2.9 AIN.VSCALE

General Information	
Type	NV Parameter
Description	Sets analog velocity scale factor.
Units	Depends on UNIT.VROTARY (pg 910) or UNIT.ACCLINEAR (pg 902) Rotary: rpm/V, rps/V, (deg/s)/V, [(Custom Units (pg 96))/s]/V, (rad/s)/V Linear: counts/s/V, (mm/s)/V, (µm/s)/V, [(Custom Units (pg 96))/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 (pg 96))/s]/V 0.006 to 6,283.186 (rad/s)/V Linear: 0.001 to 1.000 counts/s/V 0.001*MOTOR.PITCH (pg 761) to 1,000.000*MOTOR.PITCH (pg 761) (mm/s)/V 0.998*MOTOR.PITCH (pg 761) to 1,000,000.000*MOTOR.PITCH (pg 761) (µm/s)/V 0.005 to 5,000 [(Custom Units (pg 96))/s]/V
Default Value	Rotary: 0.060 rpm/V 0.001 rps/V 0.359 (deg/s)/V 0.005 [(Custom Units (pg 96))/s]/V 0.006 (rad/s)/V Linear: 0.001 counts/s/V 0.001*MOTOR.PITCH (pg 761) (mm/s)/V 0.998*MOTOR.PITCH (pg 761) (µm/s)/V 0.005 to 5,000 [(Custom Units (pg 96))/s]/V
Data Type	Float
See Also	N/A
Start Version	M_01-02-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3629h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	14	No	32 bit	No	M_01-03-00-000

Description

AIN.VSCALE is an analog velocity scale factor that scales the analog input AIN.VALUE (pg 375)) for DRV.OPMODE (pg 549) = 1 (analog velocity mode).

The value entered is the motor velocity per 1 V of analog input. This value may be either higher or lower than the application velocity limit (VL.LIMITP (pg 949) orVL.LIMITN (pg 947)), but the actual analog I/O will be limited by VL.LIMITP (pg 949) or VL.LIMITN (pg 947).

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

25.2.10 AIN.ZERO

General Information	
Type	Command
Description	Zeroes the analog input signal.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
See Also	AIN.VALUE (pg 375), AIN.OFFSET (pg 372)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	N/A	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	16	No	Command	No	M_01-03-00-000

Description

AIN.ZERO causes the drive to zero the analog input signal (AIN.VALUE (pg 375)). You may need to execute this command more than once to achieve zero offset, and AIN.OFFSET is modified in this process.

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

25.3 AIN2 Parameters

This section describes the analog input 2 (AIN2) parameters.

25.3.1 AIN2.CUTOFF

General Information	
Type	NV Parameter
Description	Sets the analog input 2 low-pass filter cutoff frequency.
Units	Hz
Range	0 to 10,000 Hz
Default Value	5,000 Hz
Data Type	Float
See Also	AIN Parameters (pg 364)
Start Version	M_01-06-03-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1234	No	32 bit	No	M_01-06-03-000

Description

AIN2.CUTOFF sets the break frequency in Hz for two cascaded single-pole low-pass filters on the hardware command input. Since the two poles are cascaded at the same frequency, the -3 dB frequency is 0.64*AIN.CUTOFF in hertz and the 10% to 90% step response rise time is 0.53/AIN2.CUTOFF in seconds.

Suggested operating values are as follows:

- Analog torque opmode: 5 kHz
- Analog velocity opmode: 2.5 kHz
- General purpose analog input high resolution: 500 Hz

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

25.3.2 AIN2.DEADBAND

General Information	
Type	NV Parameter
Description	Sets the analog input 2 signal deadband.
Units	V
Range	0 to 12.5 V
Default Value	0 V
Data Type	Float
See Also	AIN Parameters (pg 364)
Start Version	M_01-06-03-000

Variant Support

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1236	No	16 bit	No	M_01-06-03-000

Description

AIN2.DEADBAND sets the deadband of the analog input 2 signal. When AIN2.DEADBANDMODE (pg 382) is set to 0, and the value of the analog input 2 is less than the value of AIN2.DEADBAND, the analog command will be 0. When the analog input 2 is greater or equal to the AIN2.DEADBAND, then the analog command will be generated using the scaling specified.

When AIN2.DEADBANDMODE is set to 1, the analog command is 0 if the input is less than the deadband value. When the input is greater than the deadband, the output is equal to (Input - Deadband) * Scaling.

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

25.3.3 AIN2.DEADBANDMODE

General Information	
Type	NV Parameter
Description	Sets the analog input 2 deadband mode.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Integer
See Also	AIN2.DEADBAND (pg 381)
Start Version	M_01-06-03-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1238	No	16 bit	No

Description

When AIN2.DEADBANDMODE is set to 0, and the value of the analog input 2 is less than the value of AIN2.DEADBAND (pg 381), the analog command will be 0. When the analog input 2 is greater or equal to the AIN2.DEADBAND, then the analog command will be generated using the scaling specified.

When AIN2.DEADBANDMODE is set to 1, the analog command is 0 if the input is less than the deadband value. When the input is greater than the deadband, the output is equal to (Input - Deadband) * Scaling.

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309)(for the drive controller environment).

25.3.4 AIN2.MODE

General Information	
Type	NV Parameter
Description	Analog input 2 mode
Units	N/A
Range	0
Default Value	0
Data Type	Integer
See Also	AIN Parameters (pg 364)
Start Version	M_01-06-03-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1242	No	8 bit	No

AKD BASIC	
Range	0 to 1

Description

The parameter AIN2.MODE is used to assign a functionality to the voltage measured on the analog input 2 pin.

0 – The analog input value is not used by any function.

25.3.5 AIN2.OFFSET

General Information	
Type	NV Parameter
Description	Sets the analog input 2 offset.
Units	V
Range	–10 to +10 V
Default Value	0 V
Data Type	Float
See Also	AIN Parameters (pg 364), AIN2.ZERO (pg 386)
Start Version	M_01-06-03-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1244	No	16 bit	Yes	M_01-06-03-000

Description

AIN2.OFFSET sets the analog offset, which is added to the analog input 2 command to the drive. This value compensates for the analog input 2 signal (AIN.VALUE (pg 375)) offset or drift.

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

25.3.6 AIN2.VALUE

General Information	
Type	R/O Parameter
Description	Reads the value of the analog input 2 signal.
Units	V
Range	-12.5 to +12.5 V
Default Value	N/A
Data Type	Float
See Also	AIN2.OFFSET (pg 384), AIN2.ZERO (pg 386), AIN Parameters (pg 364)
Start Version	M_01-06-03-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3470h/4 3509h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1250	No	16 bit	No	M_01-06-03-000

Description

AIN2.VALUE reads the analog input 2 value after the value is filtered (as shown in the Analog Input Block Diagram).

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

25.3.7 AIN2.ZERO

General Information	
Type	Command
Description	Zeroes the analog input 2 signal.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
See Also	AIN2.VALUE (pg 385), AIN2.OFFSET (pg 384)
Start Version	M_01-06-03-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	N/A	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1258	No	Command	No	M_01-06-03-000

Description

AIN2.ZERO causes the drive to zero the analog input 2 signal (AIN2.VALUE (pg 385)). You may need to execute this command more than once to achieve zero offset, and AIN2.OFFSET (pg 384) is modified in this process.

Related Topics

Analog Input (pg 115)

Block Diagram for Position/Velocity Loop (pg 309) (for the drive controller environment).

25.4 AIO Parameters

This section describes the AIO parameters.

25.4.1 AIO.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 A/V
Data Type	Float
See Also	AIN Parameters (pg 364)
Start Version End Version	M_01-00-00-000 M_01-01-01-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1260	No	32-bit	No	M_01-06-03-000

Description

AIO.ISCALE sets the analog current scale factor that scales the following:

- The analog input (AIN.VALUE (pg 375)) for DRV.OPMODE (pg 549) = 0 (analog torque mode), DRV.CMDSOURCE = 3 (analog).
- The analog output (AOUT.VALUE (pg 401)) for AOUT.MODE (pg 396) = 5 or 6. The value entered is the motor current per 1 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 and IL.LIMITP).

25.4.2 AIO.PSCALE

General Information	
Type	NV Parameter
Description	Sets position scale factor.
Units	Depends on UNIT.PROTARY (pg 908) or UNIT.PLINEAR (pg 906) Rotary: counts/V,rad/V, deg/V, (Custom Units (pg 96))/V, 16-bit counts/V Linear: counts/V, mm/V, um/V, (Custom Units (pg 96))/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 10,737,418.240 (Custom Units (pg 96))/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 2147483.648 mm/V 0 to 2147483648.000 um/V 0 to 10737418.240 (Custom Units (pg 96))/V 0 to 140737488355.327 16-bit counts/V
Default Value	Rotary: 1 counts/V 0 rad/V 0 deg/V 0 (Custom Units (pg 96))/V 0 16-bit counts/V Linear: 1 count/V 0 rad/V 0 deg/V 0 (Custom Units (pg 96))/V 0 counts16 bit/V
Data Type	Float
See Also	Analog Input Block Diagram
Start Version End Version	M_01-00-00-000 M_01-01-01-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1262	Yes	64-bit	No	M_01-06-03-000

Description

AIO.PSCALE is an analog position scale factor that scales:

1. The analog input (AIN.VALUE (pg 375)) for DRV.OPMODE (pg 549) = 2 , DRV.CMDSOURCE (pg 497) = 3 (analog position mode)

2. The analog output (AOUT.VALUE (pg 401)) for AOUT.MODE (pg 396) = 6, or 7. (actual position or position error) per 10 V of analog input or output.

25.4.3 AIO.VSCALE

General Information	
Type	NV Parameter
Description	Sets velocity scale factor.
Units	Depends on UNIT.VROTARY (pg 910) or UNIT.ACCLINEAR (pg 902) Rotary: rpm/V, rps/V, (deg/s)/V, [(Custom Units (pg 96))/s]/V, (rad/s)/V Linear: counts/s/V, (mm/s)/V, (um/s)/V, [(Custom Units (pg 96))/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 (pg 96))/s]/V 0.006 to 6,283.186 (rad/s)/V Linear: 0.001 to 1.000 counts/s/V 0.001*MOTOR.PITCH (pg 761) to 1,000.000*MOTOR.PITCH (pg 761) (mm/s)/V 0.998*MOTOR.PITCH (pg 761) to 1,000,000.000*MOTOR.PITCH (pg 761) (um/s)/V 0.005 to 5,000 [(Custom Units (pg 96))/s]/V
Default Value	Rotary: 0.060 rpm/V 0.001 rps/V 0.359 (deg/s)/V 0.005 [(Custom Units (pg 96))/s]/V 0.006 (rad/s)/V Linear: 0.001 counts/s/V 0.001*MOTOR.PITCH (pg 761) (mm/s)/V 0.998*MOTOR.PITCH (pg 761) (um/s)/V 0.005 to 5,000 [(Custom Units (pg 96))/s]/V
Data Type	Float
See Also	Analog Input Block Diagram
Start Version End Version	M_01-00-00-000 M_01-01-01-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1266	Yes	64-bit	No	M_01-06-03-000

Description

AIO.VSCALE is an analog velocity scale factor that scales:

- The analog input (AIN.VALUE) for DRV.OPMODE = 2 (analog velocity mode)
- The analog output (AOUT.VALUE) for AOUT.MODE = 1, 3, or 7. The value entered is the motor velocity per 10 V of analog input or 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.

25.5 AOUT Parameters

This section describes the AOUT parameters.

25.5.1 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
See Also	Analog Output (pg 116)
Start Version	M_01-04-01

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1270	No	32-bit	No	M_01-06-03-000

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 116)

25.5.2 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
See Also	AOUT.VALUE (pg 401)
Start Version	M_01-01-01-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	18	No	32 bit	No	M_01-03-00-000

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 (pg 709) and IL.LIMITP (pg 710)).

Related Topics

Analog Output (pg 116)

25.5.3 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
See Also	N/A
Start Version	M_01-00-00-000

SynqNet Information	
Range	12

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3470h/1	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	20	No	16 bit	No	M_01-03-00-000

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.

AOUT.MODE	Description
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 116)

25.5.4 AOUT.OFFSET

General Information	
Type	NV Parameter
Description	Sets the analog output off-set.
Units	V
Range	-10 to +10 V
Default Value	0 V
Data Type	Float
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	22	No	16 bit	Yes	M_01-03-00-000

Description

This parameter sets the analog output offset.

Related Topics

Analog Output (pg 116)

25.5.5 AOUT.PSCALE

General Information	
Type	NV Parameter
Description	Sets the analog position scale factor.
Units	Depends on UNIT.PROTARY (pg 908) or UNIT.PLINEAR (pg 906) Rotary: counts/V,rad/V, deg/V, (Custom Units (pg 96))/V, 16-bit counts/V Linear: counts/V, mm/V, µm/V, (Custom Units (pg 96))/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 10,737,418.240 (Custom Units (pg 96))/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 µm/V 0 to 10,737,418.240 (Custom Units (pg 96)Custom Units (pg 96))/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 (Custom Units (pg 96))/V 0 16-bit counts/V Linear: 1 counts/V 0 rad/V 0 deg/V 0 (Custom Units (pg 96))/V 0 counts16 bit/V
Data Type	Float
See Also	AOUT.VALUE (pg 401)
Start Version	M_01-01-01-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3471h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	24	Yes	64 bit	No	M_01-03-00-000

Description

AOUT.PSCALE is an analog position scale factor that scales the analog output (AOUT.VALUE (pg 401)) for AOUT.MODE (pg 396) = 6, or 7 (actual position or position error) per 10 V of analog input or output.

Related Topics

Analog Output (pg 116)

25.5.6 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3470h/2	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	28	Yes	64 bit	Yes	M_01-03-00-000

Description

AOUT.VALUE reads the analog output value.

Related Topics

Analog Output (pg 116)

25.5.7 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3470h/3	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	32	Yes	64 bit	Yes	M_01-03-00-000

Description

AOUT.VALUEU reads/writes the analog output value when AOUT.MODE (pg 396) = 0 (analog output signal is determined by the user).

Related Topics

Analog Output (pg 116)

25.5.8 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
See Also	AOUT.VALUE
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3470h/5	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	36	No	32 bit	No	M_01-03-00-000

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 116)

25.6 AOUT2 Parameters

This section describes the AOUT2 parameters.

25.6.1 AOUT2.CUTOFF

General Information	
Type	NV Parameter
Description	Sets the analog output 2 low-pass filter cutoff frequency.
Units	Hz
Range	0 to 10,000 Hz
Default Value	0 Hz
Data Type	Float
See Also	Analog Output (pg 116)
Start Version	M_01-06-03-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1272	Yes	32-bit	No	M_01-06-03-000

Description

AOUT2.CUTOFF sets the cutoff frequency in Hz for a single-pole low-pass filter on the Analog Output 2.

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 2.

Related Topics

Analog Output (pg 116)

25.6.2 AOUT2.MODE

General Information	
Type	NV Parameter
Description	Sets the analog output 2 mode.
Units	N/A
Range	0
Default Value	0
Data Type	Integer
See Also	N/A
Start Version	M_01-06-03-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1276	No	16 bit	No	M_01-06-03-000

Description

AOUT2.MODE sets the analog output functionality.

Mode 0: User variable. The analog output 2 signal is determined by the user (using AOUT.VALUEU (pg 410)).

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 116)

25.6.3 AOUT2.OFFSET

General Information	
Type	NV Parameter
Description	Sets the analog output 2 offset.
Units	V
Range	-10 to +10 V
Default Value	0 V
Data Type	Float
See Also	N/A
Start Version	M_01-06-03-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1278	No	16 bit	Yes	M_01-06-03-000

Description

This parameter sets the analog output 2 offset.

Related Topics

Analog Output (pg 116)

25.6.4 AOUT2.VALUE

General Information	
Type	R/O Parameter
Description	Reads the analog output 2 value.
Units	V
Range	–10 to +10 V
Default Value	0
Data Type	Float
See Also	N/A
Start Version	M_01-06-03-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1284	Yes	64 bit	Yes	M_01-06-03-000

Description

AOUT2.VALUE reads the analog output 2 value.

Related Topics

Analog Output (pg 116)

25.6.5 AOUT.VALUEU

General Information	
Type	R/W Parameter
Description	Sets the analog output 2 value.
Units	V
Range	–10 to +10 V
Default Value	0
Data Type	Float
See Also	N/A
Start Version	M_01-06-03-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1288	Yes	64 bit	Yes	M_01-06-03-000

Description

AOUT2.VALUEU reads/writes the analog output 2 value when AOUT2.MODE (pg 407) = 0 (analog output signal is determined by the user).

Related Topics

Analog Output (pg 116)

25.7 BODE Parameters

This section describes the BODE parameters.

25.7.1 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
See Also	BODE.MODE (pg 418)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	38	No	8 bit	No	M_01-03-00-000

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 Hz) = 1/8,000 Hz = 0.000125 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 (pg 183) | Using the Performance Servo Tuner: Advanced (pg 187)

Scope (pg 228)

Settings (pg 54)

F126 (pg 251)

Error: Invalid Bode plot mode for this function. (pg 269) and other errors)

25.7.2 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
See Also	BODE.MODE (pg 418) BODE.INJECTPOINT (pg 417), BODE.IAMP, BODE.VAMP (pg 425)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	40	No	32 bit	No	M_01-03-00-000

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 (pg 183)

Using the Performance Servo Tuner: Advanced (pg 187)

Scope (pg 228)

Settings (pg 54)

F126 (pg 251)

Error: Invalid Bode plot mode for this function. (pg 269) (and others)

25.7.3 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
See Also	BODE.INJECTPOINT (pg 417), BODE.FREQ (pg 413)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	42	No	32 bit	Yes	M_01-03-00-000

Description

BODE.IAMP sets the amplitude of the excitation when in current mode as set in BODE.INJECTPOINT. When using BODE.MODE (pg 418) = 1 and BODE.INJECTPOINT (pg 417) = 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 (pg 183)

Using the Performance Servo Tuner: Advanced (pg 187)

Scope (pg 228)

Settings (pg 54)

F126 (pg 251)

Error: Invalid Bode plot mode for this function. (pg 269) (and others)

25.7.4 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
See Also	BODE.MODE (pg 418),BODE.MODETIMER (pg 422), BODE.IFTHRESH (pg 416), BODE.VFLIMIT (pg 427), BODE.VFTHRESH (pg 428)
Start Version	M_01-02-10-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1296	No	32 bit	Yes	M_01-06-03-000

Description

When BODE.MODE is set to 5, the firmware monitors the drive current (IL.CMD (pg 687)). When IL.CMD goes above BODE.IFTHRESH (pg 416), 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 (pg 183)

Using the Performance Servo Tuner: Advanced (pg 187)

F133 (pg 251)

25.7.5 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
See Also	BODE.MODE (pg 418),BODE.MODETIMER (pg 422), BODE.VFLIMIT (pg 427), BODE.VFTHRESH (pg 428) , BODE.IFLIMIT (pg 415)
Start Version	M_01-02-10-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1298	No	32 bit	Yes	M_01-06-03-000

Description

When BODE.MODE (pg 418) is set to 5, the firmware monitors the drive current (IL.CMD (pg 687)). 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 (pg 415), 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 (pg 183)

Using the Performance Servo Tuner: Advanced (pg 187)

F133 (pg 251)

25.7.6 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
See Also	BODE.IAMP (pg 414), BODE.MODE (pg 418), BODE.VAMP (pg 425)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	44	No	8 bit	No	M_01-03-00-000

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 (pg 183)

Using the Performance Servo Tuner: Advanced (pg 187)

Scope (pg 228)

Settings (pg 54)

F126 (pg 251)

Error: Invalid Bode plot mode for this function. (pg 269) and others)

25.7.7 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
See Also	BODE.INJECTPOINT (pg 417)BODE.VAMP (pg 425)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	46	No	8 bit	No	M_01-03-00-000

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.I-NJECTPOINT).

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.E-XCITEGAP drive samples. This repetition can be used to reveal the effects of friction.
2	Sine	Uses Sine excitation

BODE.MODE	Description	Comments
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

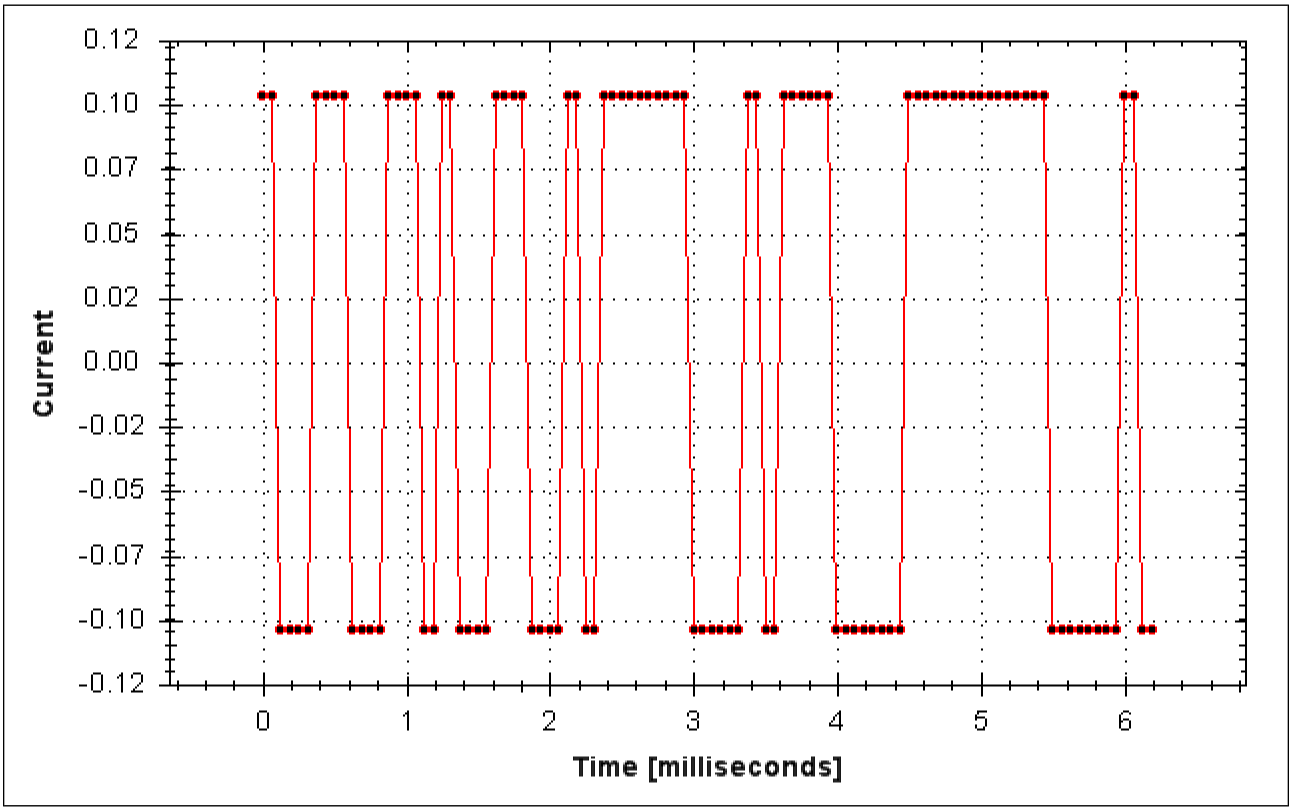
Set BODE.MODE to PRB:

```
-->BODE.MODE 1
```

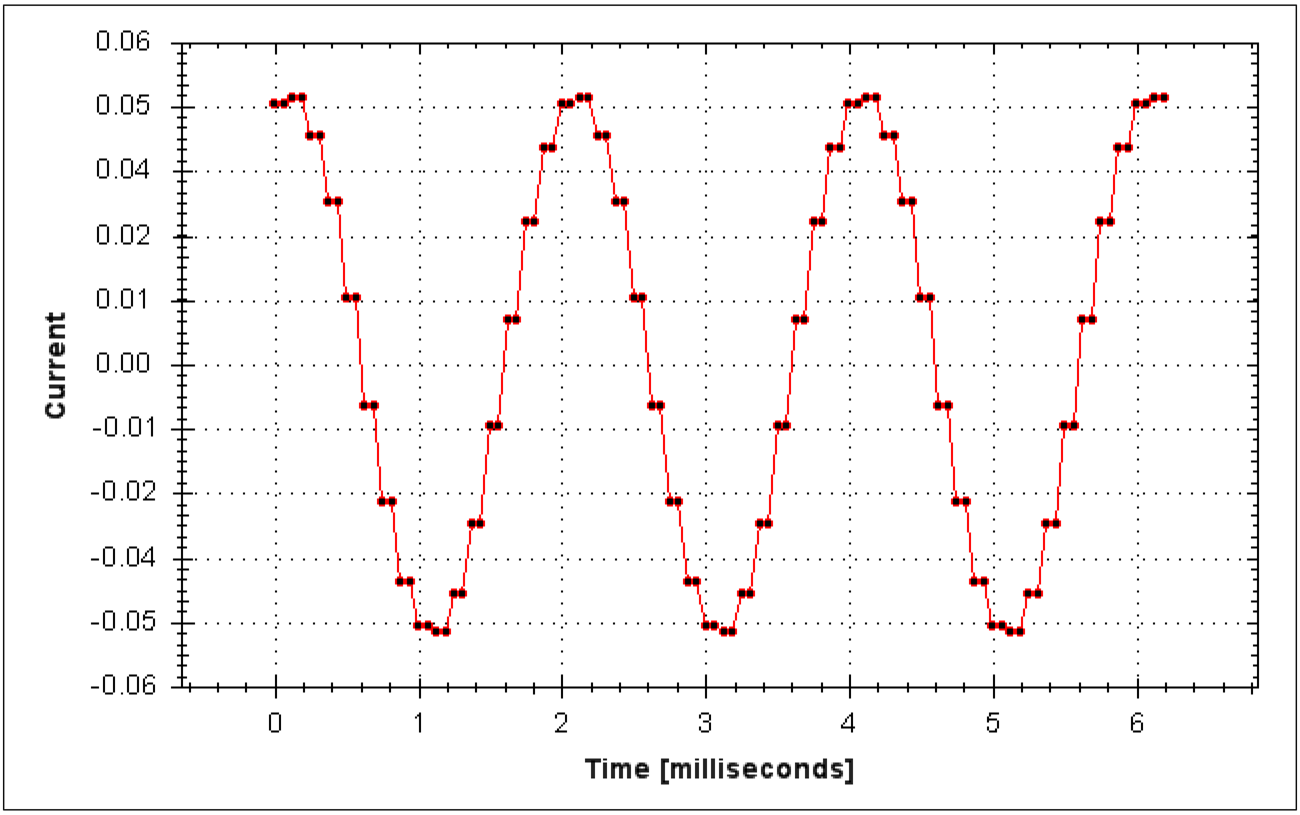
Get BODE.MODE (already set to PRB):

```
-->BODE.MODE 1
```

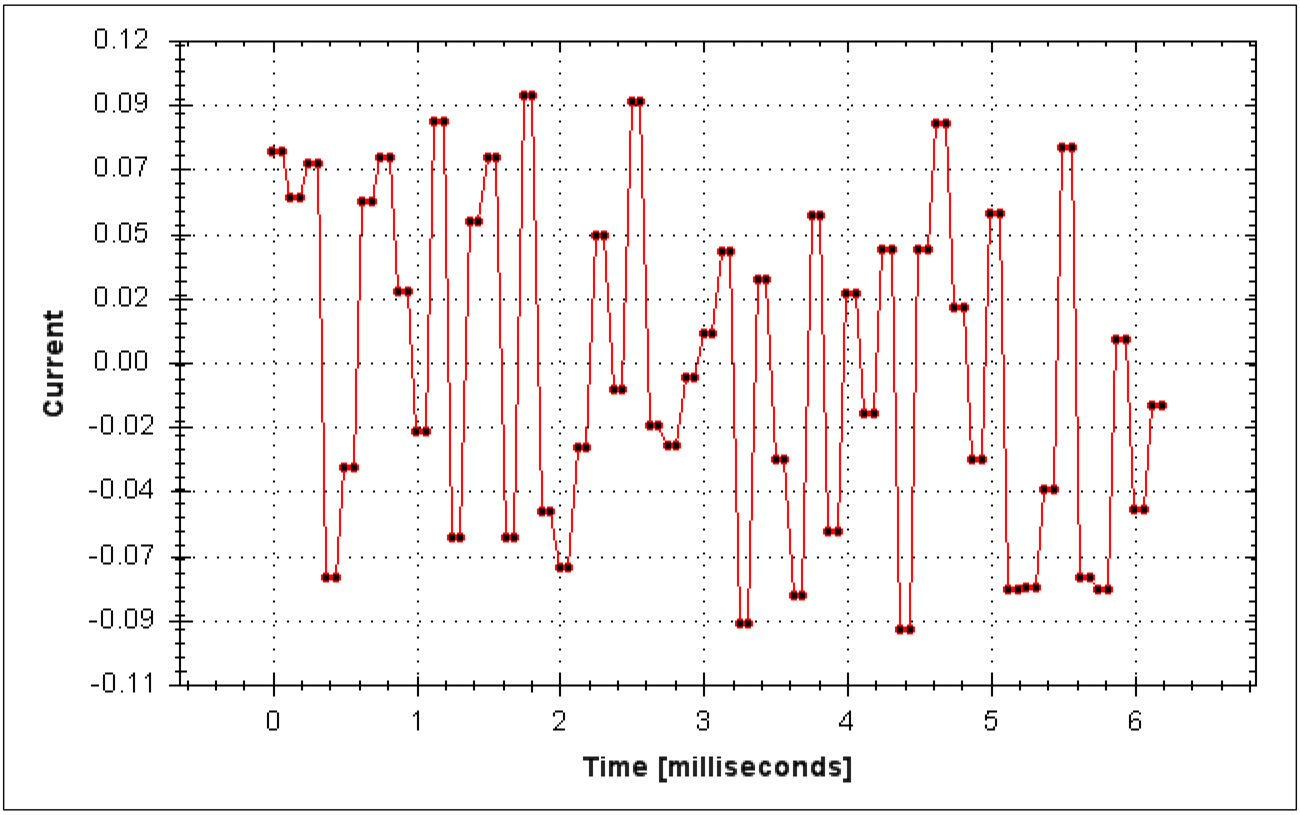
PRB excitation:



Sine excitation:



Noise excitation:



Related Topics

Using the Performance Servo Tuner (pg 183)

Using the Performance Servo Tuner: Advanced (pg 187)

Scope (pg 228)
Settings (pg 54)
F126 (pg 251)
Error: Invalid Bode plot mode for this function. (pg 269) and others)

25.7.8 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
See Also	BODE.MODE (pg 418)
Start Version	M_1-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	48	No	32 bit	No	M_01-03-00-000

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 WorkBench 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^BODE.PRBDEPTH)/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
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 (pg 183)
Using the Performance Servo Tuner: Advanced (pg 187)
Scope (pg 228)
Settings (pg 54)
F126 (pg 251)
Error: Invalid Bode plot mode for this function. (pg 269) and others)

25.7.9 BODE.PRBDEPTH

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
See Also	BODE.MODE (pg 418), BODE.INJECTPOINT (pg 417), BODE.IAMP (pg 414), BODE.VAMP (pg 425)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	50	No	8 bit	No	M_01-03-00-000

Description

BODE.PRBDEPTH sets the length of the PRB signal before it repeats. This applies only when BODE.MODE = PRB. The PRB excitation will repeat after (2^BODE.PRBDEPTH) / BODE.EXCITEGAP drive samples.

Example

Set BODE.PRBDEPTH to 19:

```
-->BODE.PRBDEPTH 19
```

Get BODE.PRBDEPTH (already set to 19):

```
-->BODE.PRBDEPTH 19
```

Related Topics

Using the Performance Servo Tuner (pg 183)

Using the Performance Servo Tuner: Advanced (pg 187)

Scope (pg 228)

Settings (pg 54)

F126 (pg 251)

Error: Invalid Bode plot mode for this function. (pg 269) and others)

25.7.10 BODE.VAMP

General Information	
Type	R/W Parameter
Description	Sets the amplitude of the excitation when in velocity mode.
Units	Depends on UNIT.VROTARY (pg 910) or UNIT.VLINEAR (pg 909)UNIT.ACCLINEAR (pg 902) Rotary: rpm, rps, deg/s, (Custom Units (pg 96))/V, rad/s Linear: counts/s, mm/s, µm/s, (Custom Units (pg 96))/V
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 1,250.000 Custom Units (pg 96)/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 (pg 96)/s
Default Value	0
Data Type	Float
See Also	BODE.MODE (pg 418), BODE.INJECTPOINT (pg 417)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	52	Yes	Low 32 bit word	Yes	M_01-03-00-000

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 (pg 183)
Using the Performance Servo Tuner: Advanced (pg 187)
Scope (pg 228)
Settings (pg 54)
F126 (pg 251)
Error: Invalid Bode plot mode for this function. (pg 269) and others)

25.7.11 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
See Also	BODE.MODE, BODE.MODETIMER, BODE.IFLIMIT, BODE.IFTHRESH, BODE.VFTHRESH
Start Version	M_01-02-10-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1300	No	32 bit	Yes	M_01-06-03-000

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 (pg 183)

Using the Performance Servo Tuner: Advanced (pg 187)

F133 (pg 251)

25.7.12 BODE.VFTHRESH

General Information	
Type	R/W Parameter
Description	Sets the current fault threshold for the BODE.MODE 5 stability test.
Units	Depends on UNIT.VROTARY (pg 910) or UNIT.VLINEAR (pg 909)UNIT.ACCLINEAR (pg 902) Rotary: rpm, rps, deg/s, (Custom Units (pg 96))/V, rad/s Linear: counts/s, mm/s, µm/s, (Custom Units (pg 96))/V
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 (pg 96)/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	0
Data Type	Decimal
See Also	BODE.MODE, BODE.MODETIMER, BODE.IFLIMIT, BODE.IFTHRESH, BODE.VFLIMIT
Start Version	M_01-02-10-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1302	Yes	64 bit	Yes	M_01-06-03-000

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 (pg 183)
Using the Performance Servo Tuner: Advanced (pg 187)
F133 (pg 251)

25.8 CAP Parameters

This section describes the CAP parameters.

25.8.1 CAP0.EDGE, CAP1.EDGE

General Information	
Type	NV Parameter
Description	Selects the capture edge.
Units	N/A
Range	1 to 3
Default Value	1
Data Type	U8
See Also	CAP0.PREEDGE, CAP1.PREEDGE (pg 439)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	54	CAP0.EDGE	No	8 bit	No	M_01-03-00-000
	80	CAP1.EDGE				

Description

The filtered trigger source is monitored for rising edge, falling edge, or both edges. The event mode logic may ignore the precondition edge detection; however, the trigger always uses edge detection.

The precondition logic has an identical feature controlled by CAP0.PREEDGE, CAP1.PREEDGE (pg 439).

Value	Description
0	Reserved
1	Rising edge
2	Falling edge
3	Both edges

Related Topics

Using Position Capture (pg 87)

25.8.2 CAP0.EN, CAP1.EN

General Information	
Type	NV Parameter
Description	Enables or disables the related capture engine.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Boolean
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	56	CAP0.EN	No	8 bit	No	M_01-03-00-000
	82	CAP1.EN				

Description

This parameter enables or disables the related capture engine. After each successful capture event, this parameter is reset to 0 and must be activated again for the next capture. Also note that CAP0.PLFB, CAP1.PLFB (pg 438) is set to 0 when this parameter is set to 1.

0 = Disable

1 = Enable

Related Topics

Using Position Capture (pg 87)

25.8.3 CAP0.EVENT, CAP1.EVENT

General Information	
Type	NV Parameter
Description	Controls the precondition logic.
Units	N/A
Range	0 to 3
Default Value	0
Data Type	U8
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	3460h/5	CAP0.EVENT	M_01-00-00-000
	3460h/6	CAP1.EVENT	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	58	CAP0.EVENT	No	8 bit	No	M_01-03-00-000
	84	CAP1.EVENT				

Description

The event mode controls use of the precondition logic. If this field is not 0, then the precondition input is selected by CAPx.TRIGGER. If this field is 1, then the precondition edge is selected by the CAPx.PREEDGE. The four event modes are listed below.

Event	Description
0	Precondition settings ignored.
1	Trigger on first trigger event after selected edge on precondition input.
2	Trigger on first trigger event to occur while precondition input is 1
3	Trigger on first trigger event to occur while precondition input is 0.

Example

Event 0

The following diagram shows an example of Event = 0 (trigger on edge, trigger edge = rising). In this mode, the precondition logic is ignored.

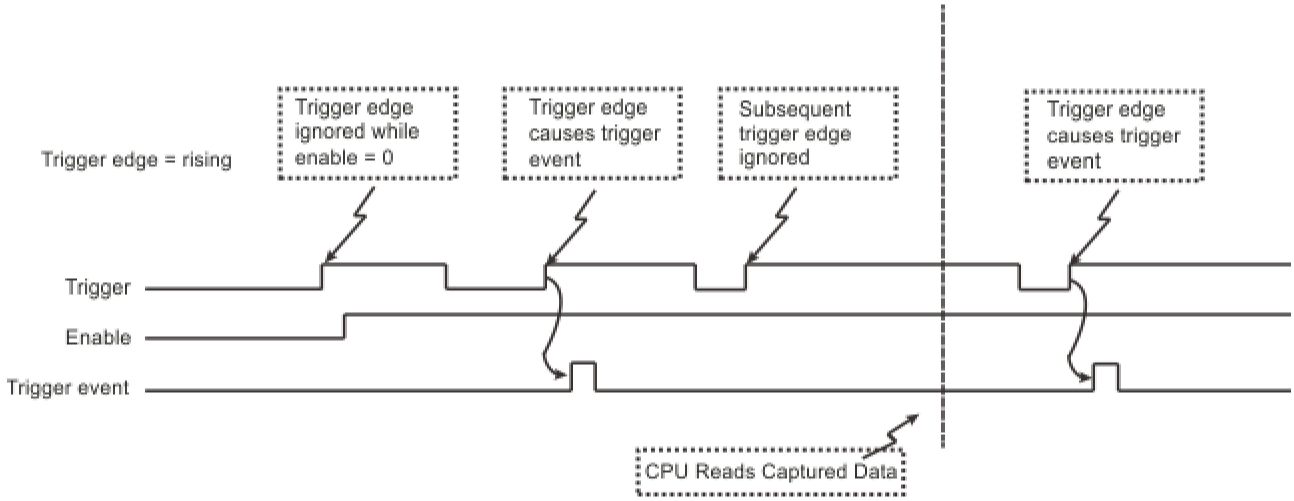


Figure 1: Trigger Edge Mode

Events 2 and 3 (Trigger edge while precondition = 0 or 1)

In these events, the precondition logic samples the current (post-filter) state of the selected precondition source input. The capture engine looks for a trigger edge while the precondition input is at a “1” or “0” state.

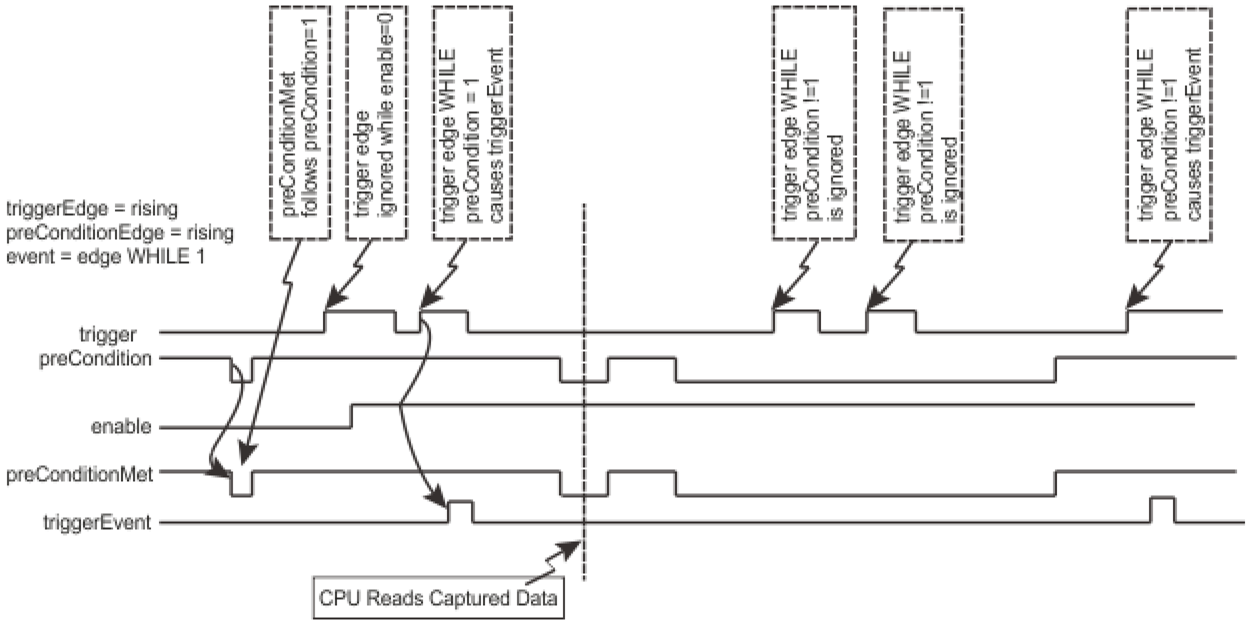


Figure 2: Trigger edge WHILE precondition edge

Event 1 (Trigger edge after precondition)

In this event, each trigger event requires Enable=1, a new precondition edge, followed by a new trigger edge. The sequence requirements are shown in the figure below.

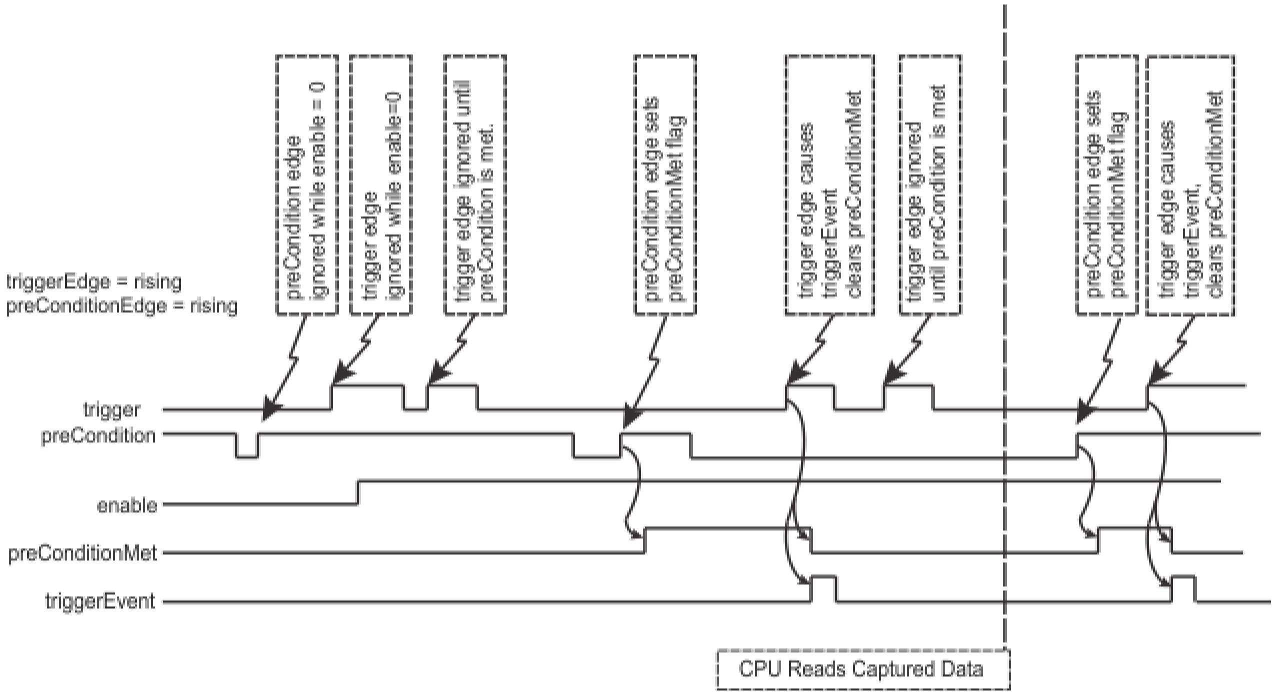


Figure 3: Trigger edge after precondition edge

Note: If the precondition and trigger edges occur at the same time, it is not a valid trigger event. A subsequent trigger edge must occur after the precondition edge. The same time resolves to a single 40 ns clock tick in the trigger event logic (after the optional filter function as well as any sensor, cable, or noise delays).

Related Topics

Using Position Capture (pg 87)

25.8.4 CAP0.FILTER, CAP1.FILTER

General Information	
Type	R/W Parameter
Description	Sets the filter for the capture source input.
Units	N/A
Range	0 to 2
Default Value	0
Data Type	U8
See Also	CAP0.PREFILTER, CAP1.PREFILTER (pg 440)
Start Version	M_01-00-00-000
End Version	M_01-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	60	CAP0.FILTER	No	8 bit	No	M_01-03-00-000
	86	CAP1.FILTER				

Description

These parameters are not functional in M_01-03-00-000. In future releases, you can use DINx.FILTER to select a filter on the input channel.

Related Topics

DIN1.FILTER TO DIN7.FILTER (pg 459)

25.8.5 CAP0.MODE, CAP1.MODE

General Information	
Type	NV Parameter
Description	Selects the captured value.
Units	N/A
Range	0 to 4
Default Value	0
Data Type	U8
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	3460h/3	CAP0.MODE	M_01-00-00-000
	3460h/4	CAP1.MODE	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	62	CAP0.MODE	No	8 bit	No	M_01-03-00-000
	88	CAP1.MODE				

Description

Mode 0 is the standard position capture, which stores PL.FB (pg 817). Data can be retrieved with CAP0.PLFB, CAP1.PLFB (pg 438).

Mode 1 is the drive internal time capture. Data can be retrieved with CAP0.T, CAP1.T (pg 444).

Mode 3 is the capture of the primary encoder signal. This mode is used to home onto a feedback index. This mode sets the other parameters needed for this mode. These parameters can be changed later, but this is not recommended unless the input source of the index signal varies. Parameters set in this mode are:

- CAPx.TRIGGER 10: index mark of primary encoder
- CAPx.EDGE 1: rising edge
- CAPx.EVENT 0: ignore precondition

Also the capture engine is immediately enabled and is continuously triggered again.

Mode 4 is similar to Mode 0 (standard position capture), except that the re-enabling of the capture is done automatically. This mode can be used for the registration move.

Related Topics

Using Position Capture (pg 87)

25.8.6 CAP0.PLFB, CAP1.PLFB

General Information	
Type	R/O Parameter
Description	Reads captured position value.
Units	Depends on UNIT.PROTARY (pg 908) or UNIT.PLINEAR (pg 906) Rotary: counts, rad, deg, Custom Units (pg 96), 16-bit counts Linear: counts, mm, µm, Custom Units (pg 96), 16-bit counts
Range	Full range of a signed 64 bit variable
Default Value	0
Data Type	S64
See Also	UNIT.PROTARY (pg 908), UNIT.PLINEAR (pg 906)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	20A0h/0	CAP0.PLFB	M_01-00-00-000
	20A1h/0	CAP0.PLFB	
	20A2h/0	CAP1.PLFB	
	20A3h/0	CAP1.PLFB	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	64	CAP0.PLFB	Yes	64 bit	Yes	M_01-03-00-000
	90	CAP1.PLFB				

Description

This parameter reads the captured position value scaled to actual set units. See UNIT.PROTARY (pg 908) or UNIT.PIN (pg 905) for these units.

Related Topics

Using Position Capture (pg 87)

25.8.7 CAP0.PREEDGE, CAP1.PREEDGE

General Information	
Type	NV Parameter
Description	Selects the capture precondition edge.
Units	N/A
Range	1 to 3
Default Value	1
Data Type	U8
See Also	CAP0.EDGE, CAP1.EDGE (pg 431)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	3460h/7	CAP0.PREEDGE	M_01-00-00-000
	3460h/8	CAP1.PREEDGE	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	68	CAP0.PREEDGE	No	8 bit	No	M_01-03-00-000
	94	CAP1.PREEDGE				

Description

The precondition edge is monitored for rising edge, falling edge, or both. The event mode logic may ignore the precondition edge detection (trigger always uses edge detection).

The filtered trigger source has an identical feature controlled by CAP0.EDGE, CAP1.EDGE (pg 431).

Value	Description
0	Reserved
1	Rising edge
2	Falling edge
3	Both edges

Related Topics

Using Position Capture (pg 87)

25.8.8 CAP0.PREFILTER, CAP1.PREFILTER

General Information	
Type	NV Parameter
Description	Sets the filter for the precondition input source.
Units	N/A
Range	0 to 2
Default Value	0
Data Type	U8
See Also	CAP0.FILTER, CAP1.FILTER (pg 436)
Start Version End Version	M_01-00-00-000 M_01-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	70	CAP0.PREFILTER	No	8 bit	No	M_01-03-00-000
	96	CAP1.PREFILTER				

Description

These parameters are not functional in M_01-03-00-000. In future releases, you can use DINx.FILTER to select a filter on the input channel.

Related Topics

DIN1.FILTER TO DIN7.FILTER (pg 459)

25.8.9 CAP0.PRESELECT, CAP1.PRESELECT

General Information	
Type	NVParameter
Description	Sets the precondition trigger.
Units	N/A
Range	0 to 11
Default Value	0
Data Type	U8
See Also	CAP0.TRIGGER, CAP1.TRIGGER (pg 445)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	3460h/9	CAP0.PRESELECT	M_01-00-00-000
	3460h/10	CAP1.PRESELECT	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	72	CAP0.PRESELECT	No	8 bit	No	M_01-03-00-000
	98	CAP1.PRESELECT				

Description

This parameter specifies the input signal for the precondition trigger.

Trigger Source	Input Name
0	General Input 1 (X7)
1	General Input 2 (X7)
2	General Input 3 (X7)
3	General Input 4 (X7)
4	General Input 5 (X8)
5	General Input 6 (X7)
6	General Input 7 (X7)
7	RS485 Input 1 (X9)

Trigger Source	Input Name
8	RS485 Input 2 (X9)
9	RS485 Input 3 (X9)
10	Primary Index

Related Topics

Using Position Capture (pg 87)

25.8.10 CAP0.STATE, CAP1.STATE

General Information	
Type	R/O Parameter
Description	Indicates whether or not trigger source was captured.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	74	CAP0.STATE	No	8 bit	No	M_01-03-00-000
	100	CAP1.STATE				

Description

When enabling the capture (CAP0.EN, CAP1.EN (pg 432)), this parameter is set to 0 until the next event is captured.

0 = Not captured or Capture Disabled

1 = Captured

Related Topics

Using Position Capture (pg 87)

25.8.11 CAP0.T, CAP1.T

General Information	
Type	R/O Parameter
Description	Reads time capture (if time capture was configured).
Units	ns
Range	N/A
Default Value	N/A
Data Type	U32
See Also	CAP0.MODE, CAP1.MODE (pg 437)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	20A0h/0	CAP0.T	M_01-00-00-000
	20A0h/0	CAP0.T	
	20A2h/0	CAP1.T	
	20A3h/0	CAP1.T	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	76	CAP0.T	No	32 bit	No	M_01-03-00-000
	102	CAP1.T				

Description

If time capture was configured, the captured time is stored in this parameter. The reference time is the occurrence of the last MTS signal (recurring every 62.5 µs), so this is a purely drive internal time.

Related Topics

Using Position Capture (pg 87)

25.8.12 CAP0.TRIGGER, CAP1.TRIGGER

General Information	
Type	NV Parameter
Description	Specifies the trigger source for the position capture.
Units	N/A
Range	0 to 11
Default Value	0
Data Type	U8
See Also	CAP0.PRESELECT, CAP1.PRESELECT (pg 441)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	3460h/1	CAP0.TRIGGER	M_01-00-00-000
	3460h/2	CAP1.TRIGGER	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	78	CAP0.TRIGGER	No	8 bit	No	M_01-03-00-000
	104	CAP1.TRIGGER				

Description

This parameter specifies the trigger source (capture input signal).

Trigger Source	Input Name
0	General Input 1
1	General Input 2
2	General Input 3
3	General Input 4
4	General Input 5
5	General Input 6
6	General Input 7

Trigger Source	Input Name
7	RS485 Input 1
8	RS485 Input 2
9	RS485 Input 3
10	Primary Index
11	Tertiary Index

Related Topics

Using Position Capture (pg 87)

25.9 CS Parameters

Controlled stop (CS) parameters set the values for the controlled stop process.

25.9.1 CS.DEC

General Information	
Type	NV Parameter
Description	Sets the deceleration value for the controlled stop process.
Units	Depends on UNIT.ACCROTARY (pg 903) or UNIT.ACCLINEAR (pg 902) Rotary: rps/s, rpm/s, deg/s^2, (Custom Units (pg 96))/s^2, rad/s^2 Linear: counts/s^2, mm/s^2, µm/s^2, (Custom Units (pg 96))/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 (pg 96))/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 (pg 761) to 833333.333*MOTOR.PITCH (pg 761) mm/s^2 30.994*MOTOR.PITCH (pg 761) to 833333333.333*MOTOR.PITCH (pg 761) µm/s^2 0.155 to 4,166,666.667 (Custom Units (pg 96))/s^2
Default Value	Rotary: 166.669 rps/s 10,000.000 rpm/s 60,000.000 deg/s^2 833.333 (Custom Units (pg 96))/s^2 1,047.2 rad/s^2 Linear: 715,840,000.000 counts/s^2 166.714*MOTOR.PITCH (pg 761)MOTOR.PITCH (pg 761) mm/s^2 166,714.191*MOTOR.PITCH (pg 761)MOTOR.PITCH (pg 761) µm/s^2 833.571 (Custom Units (pg 96))/s^2
Data Type	Float
See Also	CS.VTHRESH (pg 452), CS.TO (pg 451), DRV.DIS, DIN1.MODE TO DIN24.MODE (pg 461), DRV.DISMODE (pg 507), DRV.DISSOURCES (pg 509)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3440h/1	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	106	Yes	64 bit	No	M_01-03-00-000

Description

This parameter sets the deceleration value for the controlled stop process.

Related Topics

Controlled Stop (pg 125)

Digital Inputs and Outputs (pg 100)

Fault and Warning Messages (pg 250) (this table indicates faults for which a controlled stop occurs)

25.9.2 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
See Also	CS.DEC (pg 448), CS.VTHRESH (pg 452), CS.TO (pg 451)DRV.DISMODE (pg 507), DRV.DISSOURCES (pg 509)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3441h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	110	No	8 bit	No	M_01-03-00-000

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 (pg 125)

Digital Inputs and Outputs (pg 100)

Fault and Warning Messages (pg 250) (this table indicates faults for which a controlled stop occurs)

25.9.3 CS.TO

General Information	
Type	NV Parameter
Description	Sets the time value for the drive velocity to be within CS.VTHRESH (pg 452).
Units	ms
Range	1 to 30,000 ms
Default Value	6 ms
Data Type	Integer
See Also	CS.DEC (pg 448), CS.VTHRESH (pg 452), CS.STATE, DRV.DIS, DIN1.MODE TO DIN24.MODE (pg 461), DRV.DISMODE (pg 507), DRV.DISSOURCES (pg 509)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3440h/3	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	112	No	32 bit	No	M_01-03-00-000

Description

CS.TO is the time value for the drive velocity to be within CS.VTHRESH (pg 452) before the drive disables.

Example

Set time value to 100 ms:

```
-->CS.TO 100
```

Related Topics

Controlled Stop (pg 125)

Digital Inputs and Outputs (pg 100)

Fault and Warning Messages (pg 250) (this table indicates faults for which a controlled stop occurs)

25.9.4 CS.VTHRESH

General Information	
Type	NV Parameter
Description	Sets the velocity threshold for the controlled stop.
Units	rpm, rps, deg/s, Custom Units (pg 96)/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 (pg 96)/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 (pg 96)/s
Default Value	5 rpm
Data Type	Float
See Also	CS.DEC (pg 448), CS.TO (pg 451), CS.STATE (pg 450), DRV.DIS (pg 506), DIN1.MODE TO DIN24.MODE (pg 461), DRV.DISMODE (pg 507), DRV.DISSOURCES (pg 509)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT and CANopen	3440h/2	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	114	Yes	low 32 bit word	No	M_01-03-00-000

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 (pg 125)

Digital Inputs and Outputs (pg 100)
Fault and Warning Messages (pg 250) (this table indicates faults for which a controlled stop occurs)

25.10 DIN Parameters

This section describes the DIN parameters.

25.10.1 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
See Also	DINx.MODE, DINx.PARAM, DIN.LCMDx
Start Version	M_01-02-08-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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 (pg 112)

Digital Inputs and Outputs (pg 100)

25.10.2 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
See Also	DIN1.MODE TO DIN24.MODE (pg 461), DIN1.PARAM TO DIN7.PARAM (pg 463), DIN.HCMD1 TO DIN.HCMD4 (pg 455)
Start Version	M_01-02-08-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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 (pg 112)

Digital Inputs and Outputs (pg 100)

25.10.3 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

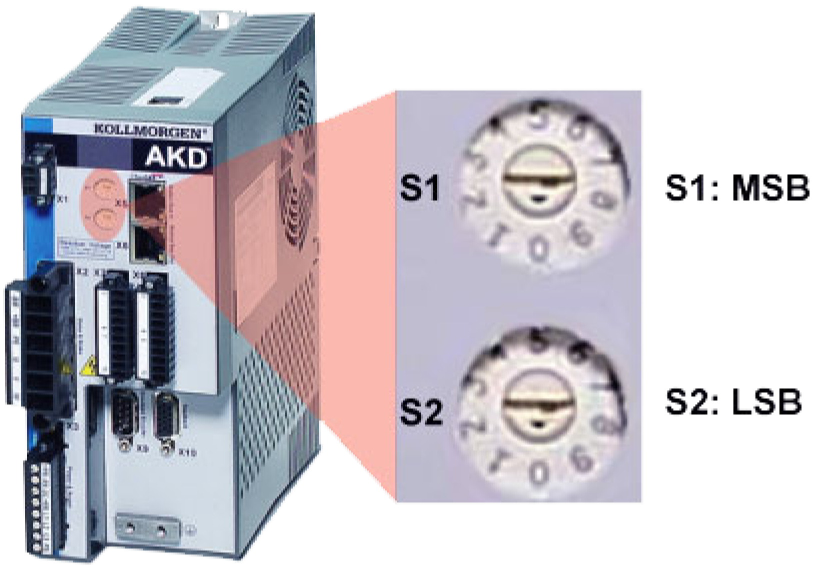
Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	116	No	8 bit	No	M_01-03-00-000

Description

DIN.ROTARY reads the rotary knob value.



Related Topics

Digital Inputs and Outputs (pg 100)

25.10.4 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Description

DIN.STATES reads the states of the seven digital inputs. The leftmost bit represents digital input 1 (DIN1) and the rightmost bit represents digital input 7 (DIN7).

Related Topics

Digital Inputs and Outputs (pg 100)

25.10.5 DIN1.FILTER TO DIN7.FILTER

General Information	
Type	R/W Parameter
Description	Filter mode for digital inputs 1 to 7.
Units	N/A
Range	0 to 3
Default Value	1 for DIN1 and DIN2 2 for DIN3 to DIN7
Data Type	Integer
See Also	N/A
Start Version	M_01-03-07-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?
Modbus	918	DIN1.FILTER	No	16 bit	No
	920	DIN2.FILTER			
	922	DIN3.FILTER			
	924	DIN4.FILTER			
	926	DIN5.FILTER			
	928	DIN6.FILTER			
	930	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 µs, ± 0.64 µs (fast filter applied).
DINX.FILTER 2	The drive digital input channel detects all input signals with an input pulse width of ≥ 163 µs, ± 10.24 µs (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 (pg 100)

25.10.6 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	120	DIN1.INV	No	8 bit	No	M_01-03-00-000
	130	DIN2.INV				
	140	DIN3.INV				
	150	DIN4.INV				
	160	DIN5.INV				
	170	DIN6.INV				
	180	DIN7.INV				

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 (pg 100)

25.10.7 DIN1.MODE TO DIN24.MODE

General Information	
Type	R/W Parameter
Description	Sets the digital input modes.
Units	N/A
Range	0 to 24
Default Value	0
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

AKD SynqNet Information	
Range	0

AKD BASIC Information	
Range	0, 1, 13, 18, 19

Variants Supported

Variant	DIN1.MODE to DIN7.MODE Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

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 DIN7.MODE	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	122	DIN1.MODE	No	16 bit	No	M_01-03-00-000
	132	DIN2.MODE				
	142	DIN3.MODE				
	152	DIN4.MODE				
	162	DIN5.MODE				
	172	DIN6.MODE				
	182	DIN7.MODE				

Description

This parameter sets the functionality of the digital inputs 1 through 7. Digital inputs and corresponding X7 and X8 pin connectors are described in the *AKD 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 100).

DINx.MODE	Description	Task
0	No function; off	0 - None
1	Fault reset	1 - Background
2	Start motion task (use DINx.PARAM for this task)	2 - 1 KHz
3	Motion task select bit (see Motion Tasks (pg 157))	3 - Background
4	Motion task start selected (see Motion Tasks (pg 157))	4 - 1 kHz
5	Start home (see Homing (pg 146))	5 - Background
6	Start jog	6 - Background
7	Reserved	7 - None
8	Zero latch	8 - Background
9	Command buffer	9 - Background
10	Control fault relay	10 - Background
11	Home reference	11 - 1 kHz
12	Reserved	12 - None
13	Controlled Stop (pg 125)	13 - 1 kHz
14	Reserved	14 - None
15	Quick stop	15 - Background
16	Activate electronic gearing (see Electronic Gearing (pg 117))	16 - Background
17	Activate electronic gear position shift	17 - Background
18	Positive limit switch	18 - 4 kHz
19	Negative limit switch	19 - 4kHz
20	Brake release	20 - Background
21	Current limit	21 - 4 kHz
22	Opmode and Command Source switch	22 - Background
23	Change algebraic sign of the measured analog input voltage.	23 - 1 kHz
24	Reserved	24 - 1 kHz
25	Controlled stop without re-enable.	25 - 1 kHz

Related Topics

Command Buffer (pg 112)

Digital Inputs and Outputs (pg 100)

Digital Inputs (pg 100)

Controlled Stop (pg 125)

Homing (pg 146)

Motion Tasks (pg 157)

Electronic Gearing (pg 117)

Clearing Faults (pg 266)

F245 (pg 256)

CS Parameters (pg 447)

25.10.8 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	124	DIN1.PARAM	Yes	64 bit	Yes	M_01-03-00-000
	134	DIN2.PARAM				
	144	DIN3.PARAM				
	154	DIN4.PARAM				
	164	DIN5.PARAM				
	174	DIN6.PARAM				
	184	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.

Input Mode	Min	Max	Notes
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 AKDs drive limits.
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 taks)

200 is larger than the maximum for DIN1.MODE 2, so DIN1.PARAM will be set to 0 to prevent errors.

Related Topics

Digital Inputs and Outputs (pg 100)

25.10.9 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	128	DIN1.STATE	No	8 bit	No	M_01-03-00-000
	138	DIN2.STATE				
	148	DIN3.STATE				
	158	DIN4.STATE				
	168	DIN5.STATE				
	178	DIN6.STATE				
	188	DIN7.STATE				

Description

DIN1.STATE to DIN7.STATE reads the state of one digital input according to the number identified in the command.

Related Topics

Digital Inputs and Outputs (pg 100)

25.10.10 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
See Also	N/A
Start Version	M_01-05-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1454	DIN9.STATE	No	8 bit	No	M_01-06-03-000
	1306	DIN10.STATE				
	1308	DIN11.STATE				

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

DIO9.DIR to DIO11.DIR (pg 471)

DIO9.INV to DIO11.INV (pg 470)

DRV.EMUEMODE (pg 514)

25.10.11 DIN21.FILTER to DIN32.FILTER

General Information	
Type	R/W Parameter
Description	Filter mode for digital inputs 21 to 32.
Units	N/A
Range	0 to 3
Default Value	2
Data Type	Integer
See Also	N/A
Start Version	M_01-03-07-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?
Modbus	1310	DIN21.FILTER	No	16 bit	No
	1322	DIN22.FILTER			
	1334	DIN23.FILTER			
	1346	DIN24.FILTER			
	1358	DIN25.FILTER			
	1370	DIN26.FILTER			
	1382	DIN27.FILTER			
	1394	DIN28.FILTER			
	1406	DIN29.FILTER			
	1418	DIN30.FILTER			
	1430	DIN31.FILTER			
	1442	DIN32.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 µs, ± 0.64 µs (fast filter applied).
DINX.FILTER 2	The drive digital input channel detects all input signals with an input pulse width of ≥ 163 µs, ± 10.24 µs (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 (pg 100)

25.10.12 DIN21.STATE to DIN32.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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?
Modbus	1320	DIN21.STATE	No	8 bit	No
	1332	DIN22.STATE			
	1344	DIN23.STATE			
	1356	DIN24.STATE			
	1368	DIN25.STATE			
	1380	DIN26.STATE			
	1392	DIN27.STATE			
	1404	DIN28.STATE			
	1416	DIN29.STATE			
	1428	DIN30.STATE			
	1440	DIN31.STATE			
	1452	DIN32.STATE			

Description

DIN21.STATE to DIN32.STATE reads the state of one digital input according to the number identified in the command.

Related Topics

Digital Inputs and Outputs (pg 100)

25.11 DIO Parameters

This section describes the DIO parameters.

25.11.1 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?
Modbus	1192	DIO10.INV	No	8 bit	No
	1196	DIO11.INV			
	1200	DIO9.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 (pg 465)

DOUT1.STATE AND DOUT2.STATE (pg 480)

DOUT1.STATEU AND DOUT2.STATEU (pg 481)

DRV.EMUEMODE (pg 514)

25.11.2 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?
Modbus	1190	DIO10.DIR	No	8 bit	No
	1194	DIO11.DIR			
	1198	DIO9.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 (pg 465)
DOUT1.STATE AND DOUT2.STATE (pg 480)
DOUT1.STATEU AND DOUT2.STATEU (pg 481)
DIO9.INV to DIO11.INV (pg 470)
DRV.EMUEMODE (pg 514)

25.12 DOUT Parameters

This section describes the DOUT parameters.

25.12.1 DOUT.CTRL

General Information	
Type	NV Parameter
Description	Sets the source of digital outputs (firmware or field-bus).
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Boolean
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	190	No	8 bit	No	M_01-03-00-000

SynqNet Information	
Range	1

Description

DOUT.CTRL sets the source of the digital outputs:

0 = Firmware controlled

1 = Fieldbus controlled

Related Topics

Digital Outputs (pg 105)

25.12.2 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	192	No	8 bit	No	M_01-03-00-000

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

Digital Outputs (pg 105)

25.12.3 DOUT.STATES

General Information	
Type	R/O Parameter
Description	Reads the state of the two digital outputs.
Units	N/A
Range	0 to 11
Default Value	N/A
Data Type	String
See Also	N/A
Start Version	M_01-00-00-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	194	No	8 bit	No	M_01-03-00-000

Description

DOUT.STATES reads the states of the two digital outputs. The rightmost bit represents DOUT2 and the leftmost bit represents DOUT1.

Related Topics

Digital Outputs (pg 105)

25.12.4 DOUT1.MODE to DOUT19.MODE

General Information	
Type	NV Parameter
Description	Sets the digital output mode.
Units	N/A
Range	0 to 19
Default Value	0
Data Type	Integer
See Also	DOUT1.PARAM AND DOUT2.PARAM (pg 478)
Start Version	M_01-04-02-000

AKD BASIC Information	
Range	0, 8, 10, 11, 15

Variants Supported

Variant	DOUT1.MODE to DOUT2.MODE Supported	DOUT3.MODE to DOUT7.MODE Supported	DOUT8.MODE TO DOUT11.MODE Supported	DOUT12.MODE to DOUT17.MODE Supported
AKD BASIC	N/A	N/A	√	N/A
AKD SynqNet	√	√	√	√
AKD EtherNet/IP	√	N/A	N/A	N/A

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	196	DOUT1.MODE	No	8 bit	No	M_01-03-00-000
	206	DOUT2.MODE				

Description

DOUTx.MODE sets the functionality of the digital outputs. The table below summarizes the digital output modes; for detailed descriptions of each mode, see Digital Inputs and Outputs (pg 100).

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

DOUTx.MODE	Description
12	Absolute velocity greater than x
13	Absolute velocity less than x
14	Homing complete
15	PLS.STATE bits or connected
16	Description Command buffer Active
17	Mt in Position
19	Encoder Z pulse
20	No Controlled Stop Active
21	Fault Disabling Power Stage

Related Topics

Digital Outputs (pg 105)

25.12.5 DOUT1.PARAM AND DOUT2.PARAM

General Information	
Type	NV Parameter
Description	Sets extra parameters for the digital outputs.
Units	N/A
Range	0 **Note:** Range changes based on Digital Output Mode. See below.
Default Value	0
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	DOUT1.PARAM Supported	DOUT2.PARAM Supported
AKD BASIC	√	√
AKD SynqNet	√	N/A
AKD EtherNet/IP	√	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	198	DOUT1.PARAM	Yes	64 bit	Yes	M_01-03-00-000
	208	DOUT2.PARAM				

Description

DOUT1.PARAM and DOUT2.PARAM set the extra parameter needed for the digital outputs calculations, respectively.

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
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.

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.

Related Topics

Digital Outputs (pg 105)

25.12.6 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	202	DOUT1.MODE	No	8 bit	No	M_01-03-00-000
	212	DOUT2.MODE				

Description

DOUT1.STATE and DOUT2.STATE read the state of one digital output according to the value stated in the command.

Related Topics

Digital Outputs (pg 105)

25.12.7 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
See Also	N/A
Start Version	M_01-01-01-000

Variants Supported

Variant	DOUT1.STATEU Supported	DOUT2.STATEU Supported
AKD BASIC	√	√
AKD SynqNet	√	N/A
AKD EtherNet/IP	√	√

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	204	DOUT1.STATEU	No	8 bit	No	M_01-03-00-000
	214	DOUT2.STATEU				

Description

DOUT1.STATEU and DOUT2.STATEU set the state of the digital output node as follows:

0 = deactivated

1 = activated

DOUT1.STATEU and DOUT2.STATEU are used when DOUT1.MODE to DOUT19.MODE (pg 476) = 0 (user mode).

Related Topics

Digital Outputs (pg 105)

25.12.8 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
See Also	N/A
Start Version	M_01-05-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1564	DOUT9.STATE	No	8 bit	No	M_01-06-03-000
	1456	DOUT10.STATE				
	1460	DOUT11.STATE				

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

DIO9.DIR to DIO11.DIR (pg 471)

DIO9.INV to DIO11.INV (pg 470)

DOUT9.STATEU to DOUT11.STATEU (pg 483)

DRV.EMUEMODE (pg 514)

25.12.9 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
See Also	N/A
Start Version	M_01-05-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1566	DOUT9.STATEU	No	8 bit	No	M_01-06-03-000
	1458	DOUT10.STATEU				
	1462	DOUT11.STATEU				

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

DIO9.DIR to DIO11.DIR (pg 471)
DIO9.INV to DIO11.INV (pg 470)
DRV.EMUEMODE (pg 514)

25.12.10 DOUT21.STATE to DOUT32.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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?
Modbus	1470	DOUT21.STATE	No	8 bit	No
	1480	DOUT22.STATE			
	1490	DOUT23.STATE			
	1500	DOUT24.STATE			
	1510	DOUT25.STATE			
	1520	DOUT26.STATE			
	1530	DOUT27.STATE			
	1540	DOUT28.STATE			
	1550	DOUT29.STATE			
	1560	DOUT30.STATE			

Description

DOUTx.STATE reads the state of one digital output according to the value stated in the command.

Related Topics

Digital Outputs (pg 105)

25.12.11 DOUT21.STATEU to DOUT32.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
See Also	N/A
Start Version	M_01-01-01-000

Variants Supported

Supported by any AKD with extended IO.

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?
Modbus	1472	DOUT21.STATEU	No	8 bit	No
	1482	DOUT22.STATEU			
	1492	DOUT23.STATEU			
	1502	DOUT24.STATEU			
	1512	DOUT25.STATEU			
	1522	DOUT26.STATEU			
	1532	DOUT27.STATEU			
	1542	DOUT28.STATEU			
	1552	DOUT29.STATEU			
	1562	DOUT30.STATEU			

Description

DOUTx.STATEU sets the state of the digital output node as follows:

0 = deactivated

1 = activated

DOUTx.STATEU is used when DOUT1.MODE to DOUT19.MODE (pg 476) = 0 (user mode).

Related Topics

Digital Outputs (pg 105)

25.13 DRV Parameters

This section describes the DRV parameters.

25.13.1 DRV.ACC

General Information	
Type	NV Parameter
Description	Describes the acceleration ramp for the velocity loop.
Units	Depends on or Rotary: rps/s, rpm/s, deg/s², (custom units)/s² , rad/s² Linear: counts/s² , mm/s² , µm/s² , (custom units)/s²
Range	Note: The range and default values of (custom units)/s² 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/s² 0.155 to 4,166,666.752 (custom units)/s² 0.012 to 5,235,987.968 rad/s² Linear: 16,000.000 to 3,579,139,408,000.000 counts/s² 0.031*MOTOR.PITCH (pg 761) to 833,333.333*MOTOR.PITCH (pg 761) mm/s² 30.995*MOTOR.PITCH (pg 761) to 2,147,483.647*MOTOR.PITCH (pg 761) µm/s² 0.155 to 2,147,483.647 (custom units)/s²
Default Value	Note: The range and default values of (custom units)/s² 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/s² 833.333 (custom units)/s² 1,047.2 rad/s² Linear: 715,840,000.000 counts/s² 166.714*MOTOR.PITCH (pg 761) mm/s² 166,714.191*MOTOR.PITCH (pg 761) µm/s² 833.571 (custom units)/s²
Data Type	Float
See Also	DRV.DEC (pg 501), ,
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3501h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	216	Yes	64 bit	No	M_01-03-00-000

Description

Describes the acceleration ramp for the velocity central loop.

Related Topics

12.6 Limits

25.13.2 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
See Also	DRV.EN (pg 520),DRV.DISSOURCES (pg 509)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	220	No	8 bit	No	M_01-03-00-000

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. Additionally, if the drive is in state 3 the Parameter Load/Save view does not allow you to download a parameter file.

If an axis is not enabled (DRV.ACTIVE is 0), but DRV.EN (pg 520) is 1 and the hardware enable is high, read the value of DRV.DISSOURCES (pg 509) to query the reason that the drive is not enabled.

Related Topics

12.8 Enable/Disable

25.13.3 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1568	No	Command	No	M_01-06-03-000

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 WorkBench.

25.13.4 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 (pg 553)

25.13.5 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	222	No	Command	No	M_01-03-00-000

Description

DRV.CLRFAULTHIST clears the fault history from the nonvolatile memory of the drive. This command erases all faults returned by DRV.FAULTHIST (pg 522).

25.13.6 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
See Also	DRV.FAULTS (pg 524), DRV.EN (pg 520), DRV.DIS (pg 506)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	224	No	Command	No	M_01-03-00-000

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 (pg 250) for details regarding the behavior of individual faults.

Note that executing a drive disable (DRV.DIS (pg 506)) followed by a drive enable (DRV.EN (pg 520)) has the same effect as executing DRV.CLRFAULTS.

Related Topics

Clearing Faults (pg 266)

25.13.7 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
See Also	N/A
Start Version	M_01-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1572	No		No	M_01-06-03-000

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.t

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
```

Related Topics

25.13.8 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
See Also	DRV.OPMODE (pg 549)
Start Version	M_01-00-00-000

AKD SynqNet Information	
Range	0

AKD BASIC Information	
Range	0, 3, 5

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	226	No	8 bit	No	M_01-03-00-000

Description

DRV.CMDSOURCE specifies the source of the command to the drive. DRV.OPMODE (pg 549) 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

25.13.9 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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.

25.13.10 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
See Also	DRV.DISMODE (pg 507)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3444h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	228	No	32 bit	No	M_01-03-00-000

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

12.9 Controlled Stop

25.9 CS Parameters

12.10 Dynamic Braking

25.13.11 DRV.DEC

General Information	
Type	NV Parameter
Description	Sets the deceleration value for the velocity loop.
Units	Depends on or Rotary: rps/s, rpm/s, deg/s², (custom units)/s², rad/s² Linear: counts/s², mm/s², µm/s², (custom units)/s²
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/s² 0.155 to 4,166,666.752 (custom units)/s² 0.012 to 5,235,987.968 rad/s² Linear: 16,000.000 to 3,579,139,408,000.000 counts/s² 0.031*MOTOR.PITCH to 833,333.333*MOTOR.PITCH mm/s² 30.994*MOTOR.PITCH to 833,333,333.333*MOTOR.PITCH µm/s² 0.155 to 4,166,666.667 (custom units)/s²
Default Value	Rotary: 166.669 rps/s 10,000.000 rpm/s 60,000.000 deg/s² 833.333 (custom units)/s² 1,047.2 rad/s² Linear: 715,840,000.000 counts/s² 166.71*MOTOR.PITCH4MOTOR.PITCH (pg 761) mm/s² 166,714.191*MOTOR.PITCHMOTOR.PITCH (pg 761) µm/s² 833.571 (custom units)/s²
Data Type	Float
See Also	DRV.ACC (pg 489), , , DRV.OPMODE (pg 549)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3522h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	230	Yes	64 bit	No	M_01-03-00-000

Description

DRV.DEC sets the deceleration value for the velocity loop command (VL.CMDU (pg 932)) and for the analog
velocity command (). The operation mode (DRV.OPMODE (pg 549)) must be set to velocity mode for this command to function.

Related Topics

12.9 Controlled Stop

12.6 Limits

25.13.12 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

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.ENCRES 0(1024)
IL.KP 50.009(24.811)
PL.KP 99.998(49.999)
VL.KP 0.108(0.000)
```

25.13.13 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	352Ah/0	M_01-00-00-000

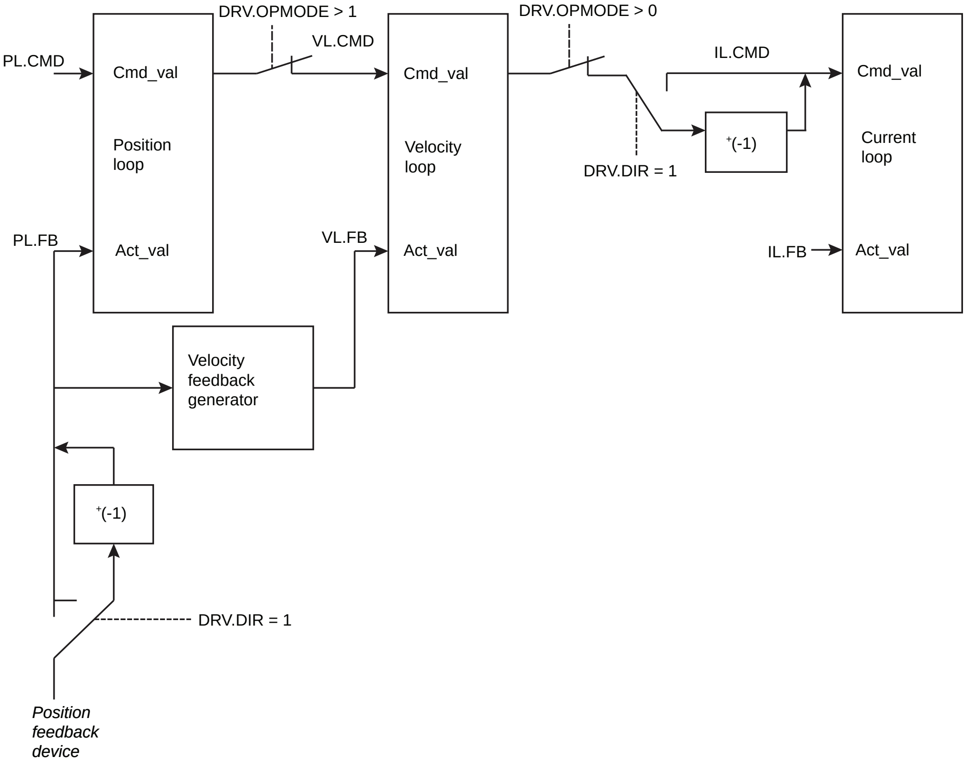
Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	234	No	8 bit	No	M_01-03-00-000

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 (pg 541)).
- 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.


DRV.OPMODE > 1
VL.CMD
DRV.OPMODE > 0
IL.CMD
PL.CMD
Cmd_val
Position
loop
Act_val
PL.FB
Cmd_val
Velocity
loop
Act_val
VL.FB
*(-1)
DRV.DIR = 1
Cmd_val
Current
loop
IL.FB
Act_val
Velocity
feedback
generator
*(-1)
DRV.DIR = 1
Position
feedback
device

25.13.14 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
See Also	DRV.EN (pg 520), DRV.DISSOURCES (pg 509), DRV.ACTIVE (pg 491), DRV.DI-SMODE (pg 507), DRV.DISTO (pg 511)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3443h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	236	No	Command	No	M_01-03-00-000

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 (pg 507).

By querying the value of DRV.ACTIVE (pg 491), you can check whether the drive is currently enabled or disabled.

By querying the value of DRV.DISSOURCES (pg 509), 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 (pg 511), fault "F703" (➜ p. 263) is reported.

Related Topics

19.4 Clearing Faults

12.9 Controlled Stop

25.13.15 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
See Also	DRV.DBILIMIT (pg 500) ,DRV.DISTO (pg 511), CS.VTHRESH (pg 452)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	35FFh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	238	No	8 bit	No	M_01-03-00-000

SynqNet Information	
Range	0 to 2

Description

DRV.DISMODE sets the drive reaction to a DRV.DIS (pg 506) command.

NOTE You must disable the drive in order to set DRV.DISMODE.

Value	Behavior
0	Disable axis immediately.
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.

In all cases described above, if a brake is configured (MOTOR.BRAKE (pg 744)), the brake closes if VL.FB (pg 934) drops below CS.VTHRESH (pg 452).

⚠WARNING Be careful with vertical loads when modifying this parameter. Coordinate this

WARNING 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.

Related Topics

12.9 Controlled Stop
25.9 CS Parameters
12.10 Dynamic Braking

25.13.16 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
See Also	DRV.ACTIVE, DRV.FAULTS, DRV.EN, DRV.DIS
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	240	No	16 bit	No	M_01-03-00-000

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
0	Software disable (execute DRV.EN to issue software enable)
1	Fault exists (read DRV.FAULTS to get the active faults)
2	Hardware disable (remote enable input is low)
3	In-rush disable (the in-rush relay is opened)
4	Initialization disable (the drive did not finish the initialization process)
5	Controlled stop disable from a digital input.
6	Field Bus requested disable (SynqNet and EtherNet/IP only)
7	AKD-C requested disable (AKD-N only)

Related Topics

Controlled Stop (pg 125)

25.13.17 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

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
7	AKD-C requested disable (AKD-N only)
8	Pre-charge successful (AKD-C only)

For most AKD models, this parameter will return the value 63.

For EtherNet/IP and SynqNet version of the drive, the parameter will return the value 127.

For AKD-N, this parameter will return the value 179.

For AKD-C, this parameter will return the value 286.

Related Topics

DRV.DISSOURCES (pg 509)

25.13.18 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
See Also	DRV.DIS (pg 506), DRV.DISMODE (pg 507)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3445h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	242	No	32 bit	No	M_01-03-00-000

Description

This timer starts when DRV.DIS (pg 506) is issued (regardless of the DRV.DIS (pg 506) origin). After this timeout elapses, the actual state of the drive is compared to the DRV.DISMODE (pg 507) setting. If the actual state does not match the DRV.DISMODE (pg 507) 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

12.9 Controlled Stop

25.13.19 DRV.EMUECHECKSPEED

General Information	
Type	TBD
Description	Enable / disable motor velocity vs. maximum emulated encoder velocity monitoring function. See Fault F486 for details.
Units	None
Range	0 to 1
Default Value	0 (disabled)
Data Type	Boolean
See Also	DRV.EMUEMODE (pg 514)
Start Version	

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1052	No	8 bit	No

Description

Enables comparison of the current motor velocity against the maximum speed the emulated encoder output can generate. The maximum speed is based on lines/rev (DRV.EMUERES) and the pulse width (DRV.EMUEPULSEWIDTH). If the motor velocity exceeds this speed, fault F486 is raised.

Related Topics

25.13.20 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
See Also	DRV.EMUEMODE (pg 514)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3493h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	244	No	8 bit	No	M_01-03-00-000

Description

This parameter allows the user to change the direction of the emulated encoder output. DRV.DIR (pg 504) also affects the output direction (through an XOR, "exclusive or", operation). The drive uses DRV.DIR (pg 504) and DRV.EMUEDIR to decide the direction of the emulated encoder output. If DRV.DIR (pg 504) 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 encoder emulation output 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 encoder emulation output and vice-versa).

25.13.21 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 14
Default Value	0
Data Type	Integer
See Also	DRV.EMUERES (pg 518), DRV.EMUEZOFFSET (pg 519), DRV.EMUEMTURN (pg 516)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3534h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	246	No	16 bit	No	M_01-03-00-000

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 and absolute index pulse (DRV.EMUEMODE is 2,7, or 9) this parameter, DRV.,EMUEZOFFSET and DRV.EMUEMTURN 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. 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

Setting	Function
2	EEO Output, A/B with absolute index pulse.
3	Input, A/B signals (Deprecated)
4	Input, step and direction signals (Deprecated)
5	Input, CW/CCW (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 absoulte Z-pulse (depends on DRV.EMUEOFFSET and DRV.EMUETURN)
10	Allows the X9 connector to be used as a General Purpose I/O or SynqNet fieldbus controlled I/O (See)
11	FB3 Input (Tertiary feedback is reported with FB3.P (pg 620)). Use FB3.MODE (pg 619) 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.

Modes 3 to 5 are backwards compatible but deprecated. Refer to FB2.MODE (pg 612) and FB2.SOURCE (pg 617) instead.

NOTE If you are using multi-turn or single tune 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

25.13.22 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
See Also	DRV.EMUEMODE (pg 514), DRV.EMUERES (pg 518)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3491h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	248	No	32 bit	No	M_01-03-00-000

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

25.13.23 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
See Also	DRV.EMUEMODE (pg 514)
Start Version	

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1050	No	32 bit	No

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
emuOutPulseWidth = (50 usecs - 40 nsec)/520 nsec = 96
actual pulse = 96*520 nsec + 40 nsec = 49.88 usecs.
```

Related Topics

25.13.24 DRV.EMUERES

General Information	
Type	R/W Parameter
Description	Sets the resolution of the EEO (emulated encoder output).
Units	lines/rev (when DRV.EMUEMODE (pg 514) = 1, 2, or 3) counts/rev (when DRV.EMUEMODE (pg 514) = 4 or 5)
Range	0 to 16,777,215 lines per revolution
Default Value	0 lines per revolution
Data Type	Integer
See Also	DRV.EMUEMODE (pg 514)
Start Version	M_01-00-00-000 (resolution increased from 65,535 to 16,777,215 in M_01-04-00-000)

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3535h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	250	No	32 bit	No	M_01-03-00-000

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

25.13.25 DRV.EMUEZOFFSET

General Information	
Type	R/W Parameter
Description	Sets the location of the EEO (emulated encoder output) index pulse (when DRV.EMU-EMODE=1).
Units	1/65536 rev
Range	0 to 65535 rev
Default Value	0 rev
Data Type	Integer
See Also	DRV.EMUEMODE (pg 514), DRV.EMUEMTURN (pg 516)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3537h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	252	No	16 bit	No	M_01-03-00-000

Description

When emulated encoder output (EEO) multiturn is selected (DRV.EMUEMODE (pg 514)=1), this parameter is used by itself to define the position if 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.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 and 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

25.13.26 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
See Also	DRV.DIS (pg 506), DRV.DISSOURCES (pg 509) DRV.ACTIVE (pg 491)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	254	No	Command	No	M_01-03-00-000

Description

DRV.EN issues a software enable to the drive. You can query the value of DRV.ACTIVE (pg 491) to check whether the drive is currently enabled or disabled.

You can also query the value of DRV.DISSOURCES (pg 509) 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

19.4 Clearing Faults

12.8 Enable/Disable

25.13.27 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	256	No	8 bit	No	M_01-03-00-000

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 (pg 123)

25.13.28 DRV.FAULTHIST

General Information	
Type	R/O Parameter
Description	Reads the last 10 faults from NV memory.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
See Also	DRV.FAULTS (pg 524), DRV.CLRFAULTHIST (pg 494)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

DRV.FAULTHISTORY 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 (pg 494) to clear this fault log.

25.13.29 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	tbd

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?
Modbus	954	DRV.FAULT1	No	16 bit	No
	956	DRV.FAULT2			
	958	DRV.FAULT3			
	960	DRV.FAULT4			
	962	DRV.FAULT5			
	964	DRV.FAULT6			
	966	DRV.FAULT7			
	968	DRV.FAULT8			
	970	DRV.FAULT9			
	972	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 (pg 524), so that fieldbusses 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

Modbus (pg 311) |DRV.ACTIVE (pg 491) | DRV.WARNING1 to DRV.WARNING10 (pg 563)

25.13.30 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
See Also	DRV.CLRFAULTS (pg 495), DRV.FAULTHIST (pg 522), DRV.CLRFAULTHIST (pg 494)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

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.
-->
```

25.13.31 DRV.HANDWHEEL

General Information	
Type	R/O Parameter
Description	Reads the EEO input value.
Units	1/4,294,967,296 rev
Range	0 to 4,294,967,295 rev
Default Value	0 rev
Data Type	Integer
See Also	DRV.EMUERES (pg 518), DRV.EMUEMODE (pg 514)
Start Version End Version	M_01-00-00-000 M_01-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	2050h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	258	No	32 bit	No	M_01-03-00-000

Description

When the EEO is selected as an input (DRV.EMUEMODE (pg 514)=3,4,5), this parameter reads the EEO value (where 4,294,967,296 is a full revolution, then the value rolls over). DRV.EMUERES (pg 518) defines the how many counts constitute a revolution on the EEO. This parameter represents the feedback 2 positions when feedback 2 is configured to be active.

When secondary feedback is selected (DRV.EMUEMODE is 0 and FB2.SOURCE = 1 (X9), or FB2.SOURCE = 2 (X7)), this parameter represents the secondary feedback position (where 4,294,967, 296 is a full revolution, then the value rolls over). FB2.ENCRES defines how many counts define a rev-olution for the secondary feedback.

Related Topics

25.13.32 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
See Also	N/A
Start Version	M_01-05-08-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1224	No	8 bit	No

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-x-xxxxx-NBxx-xxxx and will only work with Endat 2.2 multiturn encoder.

25.13.33 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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 (pg 520)) or information commands (such as DRV.VER (pg 561)).

Related Topics

Terminal (pg 240)

25.13.34 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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 (pg 520)) or pure INFO commands (such as DRV.VER (pg 561)).

Related Topics

Terminal (pg 240)

25.13.35 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1054	No	8 bit	No

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.ACITVE.

Related Topics

DRV.DISSOURCES (pg 509) | DRV.ACTIVE

25.13.36 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1222	No	8 bit	No

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 (pg 507) | CS.DEC (pg 448) | CS.VTHRESH (pg 452) | CS.TO (pg 451) | MOTOR.TBRAKEAPP (pg 765)

25.13.37 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
See Also	N/A
Start Version	M_01-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3506h/0	M_01-03-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	260	No	8 bit	No	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

12.8 Enable/Disable

25.13.38 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
See Also	DRV.IPEAK (pg 535)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	262	No	32 bit	Yes	M_01-03-00-000

Description

DRV.ICONT returns the drive continuous rated current in Arms.

25.13.39 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

DRV.INFO returns general information about the drive.

Example

```
Advanced Kollmorgen Drive
-------------------------------------------------------------------
---
Drive model : AKD-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
```

25.13.40 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
See Also	DRV.ICONT (pg 532)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	264	No	32 bit	Yes	M_01-03-00-000

Description

DRV.IPEAK returns the drive peak rated current in Arms.

Related Topics

8.6 Foldback

25.13.41 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
See Also	DRV.ZERO (pg 565)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	266	No	32 bit	No	M_01-03-00-000

Description

This parameter sets the current that is used during the DRV.ZERO (pg 565) procedure.

25.13.42 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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
```

25.13.43 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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
```

25.13.44 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
See Also	DRV.MEMDATA (pg 540)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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
```

25.13.45 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
See Also	DRV.MEMADDR (pg 539)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

DRV.MEMDATA reads a value from the address that was set by DRV.MEMADDR (pg 539) or writes a value to this address. The input value can be either decimal or hexadecimal with 0x prefix.

Examples

Read a value from internal address:

```
-->DRV.MEMDATA 01
```

Write a hexadecimal value to an internal address:

```
-->DRV.MEMADDR 0x01
```

25.13.46 DRV.MOTIONSTAT

General Information	
Type	R/O Parameter
Description	Reads the motion status of the drive.
Units	N/A
Range	0 to 4,294,967,295
Default Value	N/A
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3492h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	268	No	32 bit	No	M_01-03-00-000

Description

This command indicates the current status of the drive internal motion (see table below).

Bit	Significance	Description
0	0x00000001	Motion task is active (high active)
1	0x00000002	Home position found /reference point set (high active)
2	0x00000004	Home routine finished (high active). Bits 1 and 2 both must be set to confirm that the homing process is complete.
3	0x00000008	Homing active (high active)
4	0x00000010	Homing error condition has occurred (high active)*
5	0x00000020	Slave in electronic gearing mode synchronized (high active)
6	0x00000040	Electronic gearing is active (high active)
7	0x00000080	Emergency stop procedure in progress (high active)
8	0x00000100	Emergency stop procedure has an error (high active)
9	0x00000200	Service motion active (high active)
10	0x00000400	A motion task could not be activated /invalid MT (high active)**
11	0x00000800	Motion task target position has been reached. See also (high active).
12	0x00001000	Motion task target velocity has been reached. See also (high active).

Bit	Significance	Description
13	0x00002000	Motion task encountered an exception. A motion task exception can happen during a static motion task activation, or during activation of motion task on the fly (when velocity is not zero). The status bit will be reset automatically on successful activation of any motion, or by a command DRV.CLRFAULT.
14	0x00004000	The target position of a motion task has been crossed. This situation occurs for motion tasks with a change on the fly when triggering the DRV.STOP (pg 556) command just before the reaching the target velocity of the current active motion task. The ramp-down procedure with the motion task deceleration ramp causes the target position to be crossed (high active).

* A possible error condition for homing to a reference switch could be that no reference switch was found between two hardware limit switches.

** A possible error condition for an invalid motion task could be that a motion task tried to trigger automatically following motion task that has never been initialized (called an "empty motion" task).

25.13.47 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

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.

25.13.48 DRV.NVCHECK

General Information	
Type	R/O Parameter
Description	NV Parameter Check-sum
Units	None
Range	N/A
Default Value	N/A
Data Type	Integer
See Also	DRV.NVLIST
Start Version	M_01-04-12-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCat COE and CANopen	2019h/0	M_01-04-12-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1042	Yes	low 32 bit word	No	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 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.

25.13.49 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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.

25.13.50 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
See Also	DRV.NVLOAD DRV.NVLIST
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1576	No	Command	No	M_01-06-03-000

Description

DRV.NVLOAD loads all data from the NV memory of the drive into the RAM parameters.

25.13.51 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
See Also	DRV.RSTVAR (pg 552)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	1010h 35EBh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	938	No	Command	No

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.

25.13.52 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
See Also	Returns how long the drive has been running since first activated. (pg 553)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

This parameter returns the length of time that the drive has been running for the current session (since the last power up).

25.13.53 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
See Also	DRV.CMDSOURCE (pg 497)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	35B4h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	270	No	8 bit	No	M_01-03-00-000

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

12.9 Controlled Stop
13.3 Current Loop

13.4 Velocity Loop
13.5 Position Loop

25.13.54 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

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

25.13.55 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
See Also	
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	272	No	Command	No	M_01-03-00-000

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.

25.13.56 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1578	No		No	M_01-06-03-000

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.

25.13.57 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
See Also	DRV.SETUPREQLIST (pg 555), MOTOR.AUTOSET (pg 742)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1580	No	32-bits	No	M_01-06-03-000

Description

This parameter returns the bitwise set status of parameters that needs to be set up before the drive can be enabled. Only when this parameter returns 0 can the drive be enabled.

Parameter	Bits
IL.KP	0x00000001
MOTOR.IPEAK	0x00000002
MOTOR.ICONT	0x00000004
MOTOR.VMAX	0x00000008
MOTOR.POLES	0x00000010
MOTOR.PHASE	0x00000020

Please note that if MOTOR.AUTOSET (pg 742) 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.

25.13.58 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
See Also	DRV.SETUPREQBITS (pg 554), MOTOR.AUTOSET (pg 742)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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. Only when all commands have value of 0 can the drive be enabled.

Please note that if MOTOR.AUTOSET (pg 742) 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
-->
```

25.13.59 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	35FEh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	274	No	Command	No	M_01-03-00-000

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.

25.13.60 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3610h/0 3611h/0	M_01-00-00-000

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.
```

25.13.61 DRV.TIME

General Information	
Type	R/W
Description	A continuous time counter in the drive.
Units	Milliseconds
Range	0 to 4294967295 (~ 49 days)
Default Value	N/A
Data Type	Integer
Start Version	tbd

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1058	No	32 bit	No

Description

A continuous time counter in the drive. The timer starts at zero and counts up until it rolls over. If a new value is written to the timer, it continues to count up starting at the written value. The DRV.TIME value is set to zero when the AKD BASIC is powered on.

Related Topics

DRV.RUNTIME (pg 553) | WHEN.DRV.TIME

25.13.62 DRV.TYPE

General Information	
Type	R/O on analog, EtherCAT and CANopen models R/W on the CC drive model.
Description	Selects the operational fieldbus on CC drive models.
Units	N/A
Range	0 to 7
Default Value	2
Data Type	Integer
See Also	FBUS.TYPE (pg 638), DRV.INFO (pg 533)
Start Version	M_01-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	276	No	8 bit	No	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 the analog, EtherCAT, and CANopen drive models. To change the operational fieldbus of your drive:

1. Set DRV.TYPE to one of the following values:

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 Langauge (not fieldbus)

7 = Profinet

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 Models (pg 29)

25.13.63 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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
```

25.13.64 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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
```

25.13.65 DRV.WARNING1 to DRV.WARNING10

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	tbd

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?
Modbus	1582	DRV.WARNING1	No	16 bit	No
	1584	DRV.WARNING2			
	1586	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 fieldbuses 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 (pg 523) | Modbus (pg 311)

25.13.66 DRV.WARNINGS

General Information	
Type	R/O Parameter
Description	Reads the active warn- ings.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

DRV.WARNINGS returns a list of all currently active warnings in the system.

25.13.67 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
See Also	DRV.IZERO (pg 536)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	278	No	8 bit	No	M_01-03-00-000

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 (pg 536)). After the motor rests at this position, the commutation angle is calculated and set automatically.

25.14 EIP Parameters

This section describes the EIP parameters.

25.14.1 EIP.CONNECTED

General Information	
Type	R/O Parameter
Description	Returns state of EtherNet/IP connection.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Instance	Data Size	Data Type
EtherNet/IP	795	1 byte	Integer

Description

Returns state of EtherNet/IP connection.

Value	Description
0	Not connected
1	Connected

Related Topics

EtherNet/IP View (pg 47)

25.14.2 EIP.POSUNIT

General Information	
Type	R/W
Description	Unit scaling for Position values over EtherNet/IP.
Units	N/A
Range	0 to 4,294,967,295
Default Value	65536
Data Type	Integer
Start Version	M_01-05-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1590	No	32-bits	No	M_01-06-03-000

Description

Position values are scaled according to the Ethernet/IP Position Controller Device standard. One “Position Units” scaling value is defined, which gives the number of actual position feedback counts (at 32 bits per revolution) equal to one position unit.

From Workbench, this scaling parameter is visible in the Ethernet/IP screen or as EIP.POSUNIT in the terminal. From Ethernet/IP, this value can be accessed at attribute 0x04 Position Units of the Position Controller object.

The default value is 2^16 = 65536, which provides 2^32 / 2^16 = 2^16 counts per revolution. A value of 1 would provide 2^32 / 1 = 2^32 counts per revolution.

Related Topics

EIP.PROFUNIT (pg 569)

25.14.3 EIP.PROFUNIT

General Information	
Type	R/W
Description	Unit scaling for Velocity and Acceleration values over EtherNet/IP.
Units	N/A
Range	0 to 4,294,967,295
Default Value	65536
Data Type	Integer
Start Version	M_01-05-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1592	No	32-bits	No	M_01-06-03-000

Description

Velocity and Acceleration values are scaled according to the EtherNet/IP Position Controller Device standard. One “Profile Units” scaling value is defined, which affects both velocity and acceleration.

For velocity values, Profile Units gives the number of actual position feedback counts (at 32 bits per revolution) per second equal to one velocity unit. For acceleration values, Profile Units gives the number of actual position feedback counts (at 32 bits per revolution) per second^2 equal to one acceleration unit.

From Workbench, this scaling parameter is visible in the EtherNet/IP screen or as EIP.PROFUNIT in the terminal. From EtherNet/IP, this value can be accessed at attribute 0x05 Profile Units of the Position Controller object.

The default value is 2^16 = 65536, which provides 2^32 / 2^16 = 2^16 counts per second per revolution. A value of 1 would provide 2^32 / 1 = 2^32 counts per second per revolution.

Related Topics

EIP.POSUNIT (pg 568)

25.15 FAULT Parameters

This section describes the FAULT parameters.

25.15.1 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?
Modbus	1202	FAULT130.ACTION	No	8 bit	No
	1204	FAULT131.ACTION			
	1206	FAULT132.ACTION			
	1208	FAULT134.ACTION			
	1594	FAULT139.ACTION			
	1210	FAULT702.ACTION			
	1230	FAULT451.ACTION			

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 Ampllifier
1	Ignore (fault will not be reported)

25.16 FB1 Parameters

This section describes the FB1 parameters.

25.16.1 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 (pg 603), FB1.IDENTIFIED (pg 583)
Start Version	M_01-01-00-100 and M_01-01-03-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	280	No	8 bit	No	M_01-03-00-000

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 (pg 68)

25.16.2 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 (pg 577)

25.16.3 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3533h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	282	No	32 bit	No	M_01-03-00-000

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 mm/40 µm = 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

FB1.IDENTIFIED value	FB1.ENCRES type
33 (hiperface)	R/O
34 (biSS Mode C)	R/W
40 (Resolver)	R/W
41 (sfd)	R/O
42 (Tamagawa)	R/O

Related Topics

8.2 Feedback 1

25.16.4 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 feedback types. FB1.FAULTS will return one of the following strings as a root cause for fault 467.

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 been 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
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

Strings
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 (pg 574) if Fault 1110 is returned)
1111: Resource ID timeout

Related Topics

FB1.DIAG (pg 574)

25.16.5 FB1.HALLSTATE

General Information	
Type	R/O Parameter
Description	Reads the Hall switch values (encoder feed-back only).
Units	Binary
Range	0 0 0 to 1 1 1
Default Value	N/A
Data Type	String
See Also	N/A
Start Version	M_01-00-00-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Description

FB1.HALLSTATE reads the Hall switch values (encoder feedback only).

Related Topics

Feedback 1 (pg 68)

25.16.6 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 (pg 579)
Start Version	M_01-03-07-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	932	No	8 bit	No

Description

FB1.HALLSTATEU reads the state of Hall switch U.

Related Topics

Feedback 1 (pg 68)

25.16.7 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 (pg 579)
Start Version	M_01-03-07-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	934	No	8 bit	No

Description

FB1.HALLSTATEV reads the state of Hall switch V.

Related Topics

Feedback 1 (pg 68)

25.16.8 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 (pg 579)
Start Version	M_01-03-07-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	936	No	8 bit	No

Description

FB1.HALLSTATEW reads the state of Hall switch W.

Related Topics

Feedback 1 (pg 68)

25.16.9 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	284	No	8 bit	No	M_01-03-00-000

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
10	Incremental encoder with A/B Quad, marker pulse and Hall
11	Incremental encoder with A/B Quad, marker pulse and no Hall
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
42	Tamagawa

Related Topics

Feedback 1 (pg 68)

25.16.10 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	286	No	8 bit	Yes	M_01-03-00-000

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 (pg 68)

25.16.11 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	288	No	32 bit	No	M_01-03-00-000

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 (pg 68)

25.16.12 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

FB1.MEMVER returns the memory feedback version (only applicable for feedbacks with memory).

Related Topics

Feedback 1 (pg 68)

25.16.13 FB1.OFFSET

General Information	
Type	NV-Parameter
Description	Sets position feedback offset.
Units	Depends on UNIT.ACCROTARY or UNIT.PLINEAR Rotary: counts, rad, deg, Custom Units (pg 96), 16-bit counts Linear: counts, mm, µm, Custom Units (pg 96), 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 -5,964.390 to 5,964.390 [Custom Units (pg 96)] -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 (pg 761) to 1192.878*MOTOR.PITCH mm -1192877.952*MOTOR.PITCH to 1192877.952*MOTOR.PITCH µm -5964.390 to 5964.390 Custom Units (pg 96) -78176452.637 to 78176452.636 16-bit counts
Default Value	0
Data Type	Float
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3533h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	290	Yes	64-bits	Yes	M_01-06-03-000

Description

FB1.OFFSET is a value added to the position feedback (PL.FB (pg 817)).

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 (pg 68) | PL.FB (pg 817)

25.16.14 FB1.ORIGIN

General Information	
Type	NV Parameter
Description	Adds to the initial feedback position.
Units	Depends on or 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 counts16 Bit
Default Value	0 counts
Data Type	Float
See Also	FB1.INITSIGNED
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3656h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	294	Yes	64 bit	No	M_01-03-00-000

Description

FB1.ORIGIN is a value that is added to the feedback device position. Initial value and modulo are determined from the number of bits of the feedback:

Initial position value = (<feedback from device> + FB1.ORIGIN) modulo <number of feedback bits>

The number of feedback bits is set according to the feedback type. For memory feedbacks it is the number of feedback bits; for none memory it is always single turn.

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.

Example

This example uses set to 2 (degrees)

It also assumes that the drive is connected to a single turn feedback device with memory.

FB1.ORIGIN is set to 22 and saved into NV memory.

Drive boots and reads from feedback device position 340 degrees. According to the description section above, calculation will be:

(340 + 22) modulo 360 = 2 degrees.

Therefore the initial feedback value will be set to 2 degrees.

Related Topics

8.2 Feedback 1

25.16.15 FB1.P

General Information	
Type	R/O Parameter
Description	Reads position from the primary feedback.
Units	Depends on FB1.UNIT counts or custom units.
Range	N/A
Default Value	N/A
Data Type	S64
Start Version	M_01-05-08-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1610	Yes	64-bits	Yes	M_01-06-03-000

Description

This parameter reads the position of the primary feedback device connected to X10. The position can be read as counts or in customer 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 (pg 68) | FB1.PUNIT (pg 600) | FB1.PIN | FB1.POUT

25.16.16 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1614	No	8-bits	No	M_01-06-03-000

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

25.16.17 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 (pg 593)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	298	No	8 bit	No	M_01-03-00-000

Description

A procedure that allows the user to find the commutation angle for encoder feedback (which has no Halls).

Related Topics

Feedback 1 (pg 68)

25.16.18 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	300	No	32 bit	No	M_01-03-00-000

Description

FB1.PFINDCMDU sets the current value used during the phase finding procedure.

Related Topics

Feedback 1 (pg 68)

25.16.19 FB1.PMTBITS

General Information	
Type	R/O
Description	Returns the number of multiturn bits provided by the feedback device.
Units	Multiturn bits
Range	0 to 32
Default Value	0
Data Type	Integer
Start Version	M_01-09-00-000

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1726	No	8-bit	No

Description

This parameter returns the number of multiturn bits provided by the feedback device. If the feedback device is a singleturn device, this parameter will return 0.

Related Topics

Feedback 1 (pg 68) | FB1.PSTBITS (pg 599) | FB1.PMTSAVEEN (pg 595) | Multiturn Overflow (pg 79)

25.16.20 FB1.PMTSAVEEN

General Information	
Type	R/W
Description	Initializes memory for multiturn over-flow.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Integer
Start Version	M_01-09-00-000

Description

Initializes memory for multiturn overflow on valid feedback devices.

Related Topics

FB1.PMTBITS (pg 594) | FB1.PSTBITS (pg 599) | Multiturn Overflow (pg 79)

25.16.21 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1618	Yes	64-bits	Yes	M_01-06-03-000

Description

FB1.POFFSET is the value added to the primary feedback position (FB1.P (pg 590)).

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.

25.16.22 FB1.POLES

General Information	
Type	R/W Parameter
Description	Reads the number of feedback poles.
Units	N/A
Range	2 to 128
Default Value	2
Data Type	Integer
See Also	MOTOR.POLES
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	302	No	16 bit	No	M_01-03-00-000

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 (pg 68)

25.16.23 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	304	No	8 bit	No	M_01-03-00-000

Description

Position values transfered 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 (pg 68)

25.16.24 FB1.PSTBITS

General Information	
Type	R/O
Description	Returns the number of singleturn bits provided by the feedback device
Units	Singleturn bits
Range	0 to 32
Default Value	0
Data Type	Integer
Start Version	

Description

This parameter returns the number of singleturn bits provided by the feedback device.

Related Topics

Feedback 1 (pg 68) | FB1.PMTBITS (pg 594)

25.16.25 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1624	No	32-bits	No	M_01-06-03-000

Description

FB1.UNIT 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 (pg 590)

25.16.26 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	306	No	16 bit	No	M_01-03-00-000

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 (pg 68)

25.16.27 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	308	No	32 bit	Yes	M_01-03-00-000

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 (pg 68)

25.16.28 FB1.SELECT

General Information	
Type	NV Parameter
Description	Sets user entered type or identified type (–1).
Units	N/A
Range	-1, 10, 20, 30, 31, 32, 40, 41, 42, 46
Default Value	-1
Data Type	Integer
See Also	FB1.IDENTIFIED
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	353Bh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	310	No	8 bit	Yes	M_01-03-00-000

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

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.

Input Value	Description
10	Manually sets the type to incremental encoder. This input sets the value of FB1.IDENTIFIED to 10. If the feedback setting fails, FB1.IDENTIFIED is automatically set to 0 (no feedback identified).
20	Manually sets the type to sine encoder. This input sets the value of FB1.IDENTIFIED to 20. If the feedback setting fails, FB1.IDENTIFIED is automatically set to 0 (no feedback identified).
30	Manually sets the type to Endat 2.1. This input sets the value of FB1.IDENTIFIED to 30. If the feedback setting fails, FB1.IDENTIFIED is automatically set to 0 (no feedback identified).
31	Manually sets the type to Endat 2.2. This input sets the value of FB1.IDENTIFIED to 31. If the feedback setting fails, FB1.IDENTIFIED is automatically set to 0 (no feedback identified).
32	Manually sets the type to BiSS. This input sets the value of FB1.IDENTIFIED to 32. If the feedback setting fails, FB1.IDENTIFIED is automatically set to 0 (no feedback identified).
33	Manually sets the type to Hiperface. This input sets the value of FB1.IDENTIFIED to 33. If the feedback setting fails, FB1.IDENTIFIED is automatically set to 0 (no feedback identified). Note that all Hiperface feedback types are supported by the AKD. 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 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 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.
40	Manually sets the type to resolver. This input sets the value of FB1.IDENTIFIED to 40. If the feedback setting fails, FB1.IDENTIFIED is automatically set to 0 (no feedback identified).
41	Manually sets the type to SFD. This input sets the value of FB1.IDENTIFIED to 41. If the feedback setting fails, FB1.IDENTIFIED is automatically set to 0 (no feedback identified).
46	Manually sets the type to Hiperface DSL. This value is only supported on model numbers AKD-xxyyzz-NBxx. This value is not supported on model numbers AKD-Nxxyyz-xxyy. This value is only supported for firmware versions 01-09-00-000 and later. An external dongle between the two wires from the DSL to the X10 connector is required to use Hiperface DSL.

FB1.SELECT Feedback Types

Type	Description
0	Unknown
10	Incremental encoder with A/B Quad, marker pulse and Hall
11	Incremental encoder with A/B Quad, marker pulse and no Hall
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

Type	Description
41	SFD
42	Tamagawa
46	Hiperface DSL

Related Topics

8.2 Feedback 1

25.16.29 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	312	No	8 bit	No	M_01-03-00-000

Description

This parameter turns the tracking calibration algorithm on or off for sine-cosine or resolver.

0 = Tracking calibration is off.

1 = Tracking calibration is on.

Related Topics

Feedback 1 (pg 68)

25.16.30 FB1.USERBYTE0 to FB1.USERBYTE7

General Information	
Type	R/W
Description	Reads and writes data stored in two 32 bit words in the Endat feedback device.
Units	N/A
Range	0 to 255
Default Value	0
Data Type	Integer
Start Version	M_01-05-08-000

Description

FB1.USERBYTE, FB1.USERWORD, and FB1.USERDWORD share two 32bit words in the Endat feedback device. These three parameters offer different ways to store and access the data stored in these two 32bit 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:

<table>
<tr><td>BYTE0</td><td>BYTE1</td><td>BYTE2</td><td>BYTE3</td><td>BYTE4</td><td>BYTE5</td><td>BYTE6</td><td>BYT7</td></tr>
<tr><td colspan="2">WORD0</td><td colspan="2">WORD1</td><td colspan="2">WORD2</td><td colspan="2">WORD3</td></tr>
<tr><td colspan="4">DWORD0</td><td colspan="4">DWORD1</td></tr>
</table>

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 (pg 68) | FB1.USERWORD1 to FB1.USERWORD3 (pg 609) | FB1.USERDWORD0 to FB1.USERWORD1 (pg 608)

25.16.31 FB1.USERDWORD0 to FB1.USERWORD1

General Information	
Type	R/W
Description	Reads and writes data stored in two 32 bit words in the Endat 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

Description

FB1.USERBYTE, FB1.USERWORD, and FB1.USERDWORD share two 32bit words in the Endat feedback device. These three parameters offer different ways to store and access the data stored in these two 32bit 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:

<table>
<tr><td>BYTE0</td><td>BYTE1</td><td>BYTE2</td><td>BYTE3</td><td>BYTE4</td><td>BYTE5</td><td>BYTE6</td><td>BYT7</td></tr>
<tr><td colspan="2">WORD0</td><td colspan="2">WORD1</td><td colspan="2">WORD2</td><td colspan="2">WORD3</td></tr>
<tr><td colspan="4">DWORD0</td><td colspan="4">DWORD1</td></tr>
</table>

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 (pg 68) | FB1.USERBYTE0 to FB1.USERBYTE7 (pg 607) | FB1.USERWORD1 to FB1.USERWORD3 (pg 609)

25.16.32 FB1.USERWORD1 to FB1.USERWORD3

General Information	
Type	R/W
Description	Reads and writes data stored in two 32 bit words in the Endat feedback device.
Units	N/A
Range	0 to 65,535
Default Value	0
Data Type	Integer
Start Version	M_01-05-08-000

Description

FB1.USERBYTE, FB1.USERWORD, and FB1.USERDWORD share two 32bit words in the Endat feedback device. These three parameters offer different ways to store and access the data stored in these two 32bit 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:

<table>
<tr><td>BYTE0</td><td>BYTE1</td><td>BYTE2</td><td>BYTE3</td><td>BYTE4</td><td>BYTE5</td><td>BYTE6</td><td>BYT7</td></tr>
<tr><td colspan="2">WORD0</td><td colspan="2">WORD1</td><td colspan="2">WORD2</td><td colspan="2">WORD3</td></tr>
<tr><td colspan="4">DWORD0</td><td colspan="4">DWORD1</td></tr>
</table>

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 (pg 68) | FB1.USERBYTE0 to FB1.USERBYTE7 (pg 607) | FB1.USERDWORD0 to FB1.USERWORD1 (pg 608)

25.17 FB2 Parameters

This section describes the FB2 parameters.

25.17.1 FB2.ENCRES

General Information	
Type	NV Parameter
Description	Sets the secondary feedback (FB2) resolution (also defines resolution of virtual encoder in AKD BASIC).
Units	counts/rev
Range	0 to 262,140 counts/rev
Default Value	0
Data Type	Integer
See Also	FB2.MODE, FB2.SOURCE (pg 617)
Start Version	M_01-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	984	No	32 bit	No

Description

This parameter sets the feedback 2 (FB2) resolution and defines how many counts input into the secondary feedback will be considered a full revolution.

In AKD BASIC FB2.ENCRES also defines the resolution of the virtual encoder when using the CAMVM commands. A runtime error will occur if the you use the CAMVM commands without defining a resolution in FB2.ENCRES. If a real encoder and a virtual encoder are both used, the resolution for the virtual encoder will be that of the real encoder.

Related Topics

Feedback 2 (pg 76)

25.17.2 FB2.MODE

General Information	
Type	R/W Parameter
Description	Sets the mode for the second feedback inputs, EEO connector (X9) and high speed opto inputs (pins 9 and 10 on X7).
Units	N/A
Range	0 to 2
Default Value	0
Data Type	Integer
See Also	FB2.ENCRES (pg 611), PL.FBSOURCE (pg 818)
Start Version	M_01-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	986	No	16 bit	No

Description

This parameter sets the feedback 2 input type as follows:

0 = Input A/B Signals

1 = Input Step and Direction Signals

2 = Input, up-down signals

Related Topics

Feedback 2 (pg 76)

FB2.SOURCE

25.17.3 FB2.P

General Information	
Type	R/O Parameter
Description	Reads position from the secondary feedback.
Units	Depends on FB2.UNIT counts or custom units.
Range	N/A
Default Value	N/A
Data Type	U64
See Also	FB1.HALLSTATE (pg 579)
Start Version	M_01-05-08-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1632	Yes	64-bits	Yes	M_01-06-03-000

Description

This parameter reads the position back from the secondary feedback device that is connected to X7 or X9, depending on the value of DRV.EMUEMODE. The position can be read as 32-bit counts or in customer units.

Related Topics

Feedback 1 (pg 68) | DRV.EMUEMODE (pg 514) | FB2.PUNIT (pg 616) | FB2.PIN | FB2.POUT | FB2.DIR (pg 614)

25.17.4 FB2.DIR

General Information	
Type	NV-Parameter
Description	Sets the counting direction for feedback channel 2.
Units	None
Range	0 to 1
Default Value	0
Data Type	U8
See Also	N/A
Start Version	M_01-05-11-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Description

FB2.DIR will change the sign and with it the direction of feedback channel 2.

25.17.5 FB2.POFFSET

General Information	
Type	NV-Parameter
Description	Sets the offset for secondary 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1638	Yes	64-bits	Yes	M_01-06-03-000

Description

FB2.POFFSET is the value added to the primary feedback position (FB2.P (pg 613)).

Example

If FB2.P is 10000 counts and FB2.POFFSET is set to –10000 counts, then the next read of FB2.P will return ~0 counts.

25.17.6 FB2.PUNIT

General Information	
Type	NV Parameter
Description	Sets the unit for FB2.P.
Units	N/A
Range	0, 3
Default Value	0
Data Type	U8
See Also	N/A
Start Version	M_01-05-11-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1644	No	32-bits	No	M_01-06-03-000

Description

FB2.UNIT sets the position unit for FB2.P.

Value	Description
0	Counts (32 bit format)
3	(FB2.PIN/FB2.POUT) per revolution.

Related Topics

FB2.P (pg 613)

25.17.7 FB2.SOURCE

General Information	
Type	R/W Parameter
Description	Sets the source for the second feedback input. Choices are the EEO connectors (X9) which are RS485 inputs, or the X7 connector's high speed opto inputs (pins 9 and 10).
Units	N/A
Range	0 to 2
Default Value	0
Data Type	Integer
See Also	FB2.ENCRES, FB2.MODE, PL.FBSOURCE (pg 818)
Start Version	M_01-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	988	No	16 bit	No

Description

This parameter sets the secondary feedback source to be either the EEO connector (X9) or the high speed opto inputs on the I/O Connector (X7) as follows:

0 = None

1 = Feedback Source X9 (EEO connector)

2 = Feedback Source X7 (High Speed Opto Inputs on the I/O Connector)

FB2 5V Power Supply Notes:

If FB2.SOURCE = 0 or 2 (X7), the 5V power supply will remain off.

If FB2.SOURCE = 1 (X9) the FB2 5V power supply will remain on unless a short is detected. This is useful for encoder types that may require calibration.

Related Topics

Feedback 2 (pg 76)

25.18 FB3 Parameters

This section describes the FB3 parameters.

25.18.1 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex
Modbus	1044

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-x-xxxxx-NBxx-xxxx.

25.18.2 FB3.P

General Information	
Type	R/O Parameter
Description	Reads position from the tertiary feedback.
Units	Depends on FB3.UNIT 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1646	Yes	64-bits	Yes	M_01-06-03-000

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 singed 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-x-xyyzz-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 (pg 514) | FB3.PUNIT (pg 623) | FB3.PIN | FB3.POUT | FB3.PDIR (pg 621) |

25.18.3 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1650	No	8-bits	No	M_01-06-03-000

Description

FB3.PDIR will change the sign and with it the direction of feedback channel 3.

Example

If position feedback = 35,185.932 and you set:

--> FB3.PDIR 1

then position feedback = -35,185.932

25.18.4 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1654	Yes	64-bits	Yes	M_01-06-03-000

Description

FB3.POFFSET is the value added to the primary feedback position (FB3.P (pg 620)).

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.

25.18.5 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1660	No	32-bits	No	M_01-06-03-000

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 (pg 620)

25.19 FBUS Parameters

This section describes the FBUS parameters.

25.19.1 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

Variants Supported

Variant	FBUS.PARAM1 to FBUS.PARAM7 Supported	FBUS.PARAM8 to FBUS.PARAM20 Supported
AKD BASIC	N/A	N/A
AKD SynqNet	√	√
AKD EtherNet/IP	√	N/A

Fieldbus Information

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	

Parameter	Range	
	EtherCAT COE	CANopen
FBUS.PARAM01	N/A	125; 250; 500; 1000
FBUS.PARAM02	0 to 1	0 to 1
FBUS.PARAM03	1,001 to 65,535	N/A
FBUS.PARAM04	0 to 1	0 to 1
FBUS.PARAM05	0 to 511	0 to 511

Parameter	Default Value	
	EtherCAT COE	CANopen
FBUS.PARAM01	N/A	125
FBUS.PARAM02	1	0
FBUS.PARAM03	0	N/A
FBUS.PARAM04	1	0
FBUS.PARAM05	0	0

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	FBUS.PARAM01	314	No	32 bit	No	M_01-03-00-000
	FBUS.PARAM02	316				
	FBUS.PARAM03	318				
	FBUS.PARAM04	320				
	FBUS.PARAM05	322				
	FBUS.PARAM06	324				
	FBUS.PARAM07	326				
	FBUS.PARAM08	328				
	FBUS.PARAM09	330				
	FBUS.PARAM10	332				

Description

FBUS.PARAM01 sets the baud rate for the CANbus. Supported baud rates are 125, 250, 500 and 1000 kBaud.

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 the configured station alias for EtherCAT.

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.

When more than three CANopen sync messages or seven EtherCAT frames have not been received, and the drive is enabled, fault F125 (pg 251), "Synchronization lost", occurs.

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.
- **Bit 0 = 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.
- **Bit 1 = 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**: Workbench/Telnet can software enable the drive when CANopen/EtherCAT are operational.
- **Bit 2 = 0**: Workbench/Telnet can switch the DS402-state machine to the state “Switched On” (enable the power stage), if the former state was “Switched on Disabled.” This requires the hardware enable to be set to 1.
- **Bit 3 = 1**: DS402-state machine is not influenced, if the software-enable is taken away via Telnet.
- **Bit 3 = 0**: DS402-state machine is influenced, if the software-enable is taken away via Telnet.
- **Bit 4 = 1**: Position scaling for DS402-parameters is done dependent on DS402-scaling parameters via objects 6091 / 6092. Velocity scaling via 204c.
- **Bit 4 = 0**: Position scaling for DS402-parameters is done dependent on UNIT.PIN.
- **Bit 5 = 1**: EtherCAT: The setting of the rotary switches define the station alias address, if FBUS. PARAM03 is on 0 else FBUS.PARAM03 is used to store the station alias address.
- **Bit 5 = 0**: EtherCAT: The setting of the rotary switches define the station alias address. If the setting is 0, FBUS.PARAM03 can be used.
- **Bit 6 = 1**: Bit 0 of the parameter MT.CNTL (object 35D9 sub 0) can be accessed.
- **Bit 6 = 0**: Bit 0 of the parameter MT.CNTL (object 35D9 sub 0) is exclusively used for DS402 control word.
- **Bit 7**: Reserved
- **Bit 8 = 1**: DS402-state switched on, power stage disabled.
- **Bit 8 = 0**: DS402-state switched on, power stage enabled.

25.19.2 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 (pg 625), Fieldbus documentation
Start Version	M_01-02-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

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

For more information, see the *AKD CANopen Manual*, Phase Locked Loop.

25.19.3 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	354	No	16 bit	No	M_01-03-00-000

Description

This parameter sets number of successful synchronized cycles needed to lock the PLL.

25.19.4 FBUS.PROTECTION

General Information	
Type	R/W Parameter
Description	Controls which parameters are blocked from being accessed through telnet while a field-bus is operational.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Integer
Start Version	M_01-08-00-000

Variants Supported

Variant	Supported
AKD SynqNet	No
AKD EtherNet/IP	No
AKD CANopen	Yes
AKD EtherCAT	Yes
AKD Profinet	No

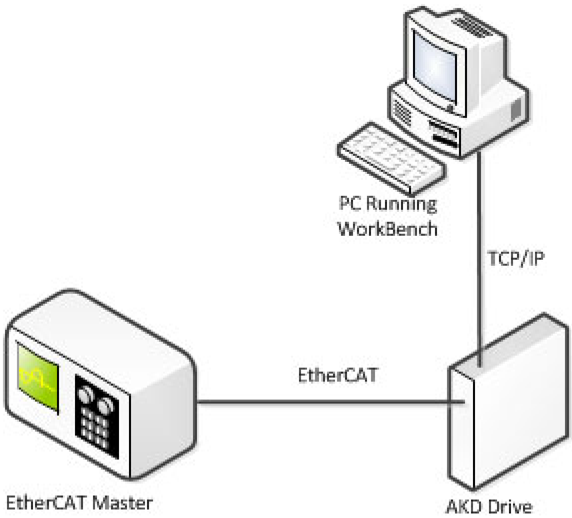
Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3498h/0	M_01-08-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1742	No	8 bit	No	M_01-08-00-000

Description

When an AKD is a slave node on a fieldbus (for example an AKD is connected to an EtherCAT master) and WorkBench is simultaneously connected to an AKD via a service channel, then you have several options on how you would like the AKD to respond to commands from each of the channels.



When the fieldbus is operational the AKD uses the FBUS.PROTECTION parameter to allow the user to select how commands from WorkBench, or any other device sending commands to the drive through a service channel, should be handled. WorkBench 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.
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

25.19.5 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
See Also	Fieldbus Manuals (pg 310)PL.FBSOURCE (pg 818)
Start Version	M_01-05-06-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

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.

25.19.6 FBUS.SAMPLEPERIOD

General Information	
Type	NV Parameter
Description	Sets fieldbus sample period.
Units	Whole multiples of MTS 250 µs
Range	4 to 128 and value must be a power of 4
Default Value	32 = 2 ms
Data Type	U8
See Also	Appendix B: Fieldbus Manuals
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	60C2h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	356	No	8 bit	No	M_01-03-00-000

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 us (4), 500 us (8), 1 ms (16), 2 ms (32), 4 ms (64).

25.19.7 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

Variants Supported

Variant	Supported
AKD Analog	N/A
AKD CANopen via CAN	√
AKD EtherCAT (CANopen)	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A
AKD Profinet	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1746	No	8-bit	Yes

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	-

25.19.8 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	358	No	32 bit	No	M_01-03-00-000

Description

This parameter reads actual distance from the desired sync distance.

25.19.9 FBUS.SYNCDIST

General Information	
Type	NV Parameter
Description	Sets time target for syn-chronization.
Units	ns
Range	0 to 250,000 ns
Default Value	100,000 ns
Data Type	Integer, U32
See Also	Appendix B: Fieldbus Manuals
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	360	No	32 bit	No	M_01-03-00-000

Description

This parameter sets time target for synchronization.

25.19.10 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	362	No	32 bit	No	M_01-03-00-000

Description

This parameter sets symmetrically arranged window around the desired sync distance.

25.19.11 FBUS.TYPE

General Information	
Type	R/O Parameter
Description	Shows the active fieldbus type.
Units	N/A
Range	0 to 5
Default Value	0
Data Type	U8
See Also	Fieldbus Manuals
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	364	No	8 bit	No	M_01-03-00-000

Description

FBUS.TYPE shows the active fieldbus type.

0 = Analog

1 = SynqNet

2 = EtherCAT

3 = CANopen

4 = EthernetIP

5 = Profinet

25.20 GEAR Parameters

This section describes the GEAR parameters.

25.20.1 GEAR.ACCMAX

General Information	
Type	NV Parameter
Description	Sets the maximum allowed acceleration value; active in opmode 2 (position) only.
Units	Depends on UNIT.ACCROTARY (pg 903) or UNIT.ACCLINEAR (pg 902) Rotary: rps/s, rpm/s, deg/s^2, (Custom Units (pg 96))/s^2, rad/s^2 Linear: counts/s^2, mm/s^2, µm/s^2, (Custom Units (pg 96))/s^2
Range	Rotary: 0.004 to 4,000,000 rps/s 0.224 to 240,000,008.192 rpm/s 1.341 to 1,439,999,983.616 deg/s^2 0.023 to 25,132,740.608 rad/s^2 Linear: 16,000.000 to 3,579,139,408,000.000 counts/s^2 0.031*MOTOR.PITCH (pg 761) to 833,333.333*MOTOR.PITCH (pg 761) mm/s^2 30.994*MOTOR.PITCH (pg 761) to 83,3333,333.333*MOTOR.PITCH (pg 761) µm/s^2
Default Value	Rotary: 166.669 rps/s 10,000.170 rpm/s 60,001.016 deg/s^2 1,047.215 rad/s^2 Linear: 715,840,000.000 counts/s^2 166.714*MOTOR.PITCHMOTOR.PITCH (pg 761) mm/s^2 166,714.191*MOTOR.PITCHMOTOR.PITCH (pg 761) µm/s^2
Data Type	Float
See Also	UNIT.ACCROTARY (pg 903), UNIT.ACCLINEAR (pg 902), GEAR.DECMAX (pg 641)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	366	Yes	64 bit	No	M_01-03-00-000

Description

This parameter limits the acceleration of the slave to a numerical higher value.

Related Topics

Electronic Gearing (pg 117)

25.20.2 GEAR.DECMAX

General Information	
Type	NV Parameter
Description	Sets the maximum allowed deceleration value; active in opmode 2 (position) only.
Units	Depends on UNIT.ACCROTARY (pg 903) or UNIT.ACCLINEAR (pg 902) Rotary: rps/s, rpm/s, deg/s^2, (Custom Units (pg 96))/s^2, rad/s^2 Linear: counts/s^2, mm/s^2, µm/s^2, (Custom Units (pg 96))/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 (pg 96))/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 (pg 761) to 833,333.333*MOTOR.PITCH (pg 761) mm/s^2 30.994*MOTOR.PITCH (pg 761) to 833,333,333.333*MOTOR.PITCH (pg 761) µm/s^2 0.155 to 4,166,666.667 (Custom Units (pg 96))/s^2
Default Value	Rotary: 166.669 rps/s 10,000.000 rpm/s 60,000.000 deg/s^2 833.333 (Custom Units (pg 96))/s^2 1,047.2 rad/s^2 Linear: 715,840,000.000 counts/s^2 166.714*MOTOR.PITCHMOTOR.PITCH (pg 761) mm/s^2 166,714.191*MOTOR.PITCHMOTOR.PITCH (pg 761) µm/s^2 833.571 (Custom Units (pg 96))/s^2
Data Type	Float
See Also	UNIT.ACCROTARY (pg 903), UNIT.ACCLINEAR (pg 902), GEAR.ACCMAX (pg 640)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	370	Yes	64 bit	No	M_01-03-00-000

Description

This parameter limits the deceleration of the slave to a numerical higher value.

Related Topics

Electronic Gearing (pg 117)

25.20.3 GEAR.IN

General Information	
Type	NV Parameter
Description	Sets the denominator of the electronic gearing ratio; active in opmode 2 (position) only.
Units	N/A
Range	1 to 65,535
Default Value	1
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	374	No	16 bit	No	M_01-03-00-000

Description

This parameter sets the denominator of the gear ratio for the electronic gearing mode. The gear ratio is used in order to increase and decrease the slave velocity. The slave velocity can be calculated by the following formula:

Slave velocity = Master velocity * GEAR.OUT (pg 647)/GEAR.IN

Be sure that you set the external master source number of signals per revolution correctly. Also, select the gear ratio so that the maximum electronic gearing velocity (GEAR.VMAX (pg 648)) is not exceeded.

Master velocitymax * GEAR.OUT (pg 647)/GEAR.IN < GEAR.VMAX (pg 648)

Related Topics

Electronic Gearing (pg 117)

25.20.4 GEAR.MODE

General Information	
Type	NV Parameter
Description	Selects electronic gearing mode; active in opmode 2 (position) only.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

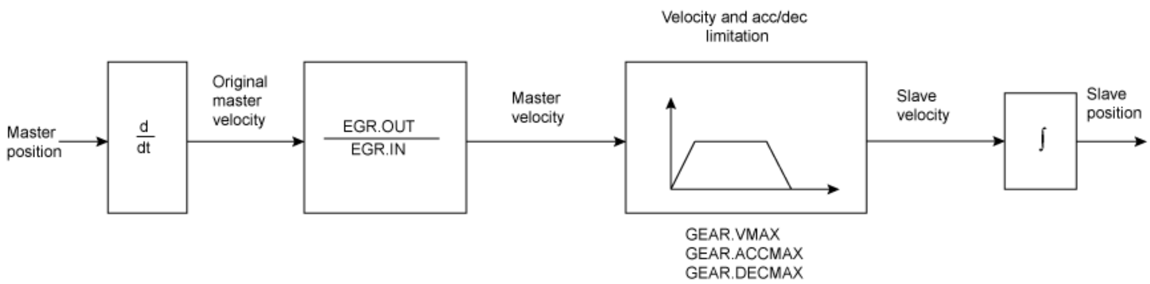
Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	376	No	16 bit	No	M_01-03-00-000

Description

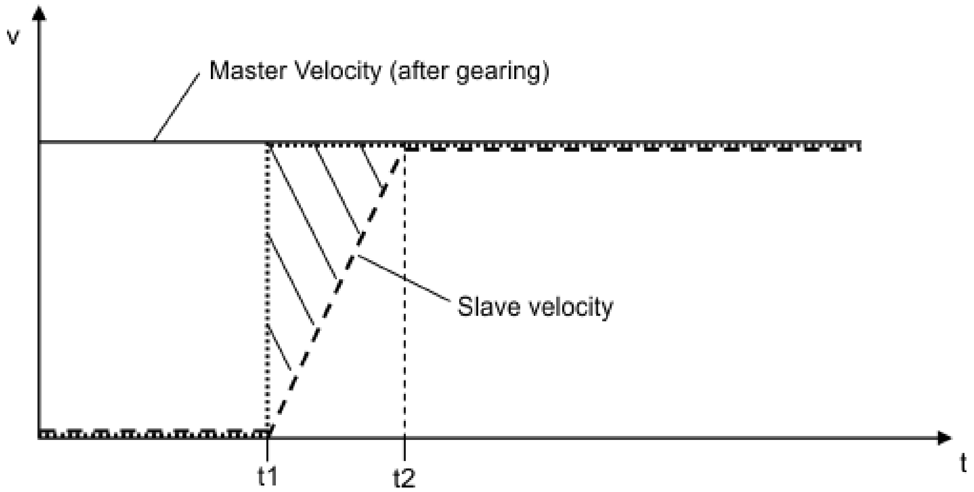
This parameter selects the electronic gearing mode at the beginning of the electronic gearing procedure. The gearing mode determines drive synchronization. In the context of electronic gearing, synchronization means that the slave follows the master pulses without losing counts due to acceleration or velocity limitations.

- Mode 0: The slave is not immediately synchronized after a GEAR.MOVE command. The slave accelerates until the master velocity (velocity after the gearing) has been reached. The drive is synchronized as soon as the velocity of the master has been reached.
- Mode 1: The slave is immediately synchronized after a GEAR.MOVE command.



Block diagram of the electronic gearing feature

The slave position is forwarded to the position loop. Ensure that DRV.OPMODE has been set to 2 and DRV.CMDSOURCE has been set to 2. The slave is able to reach the master velocity according to the GEAR.VMAX setting. GEAR.VMAX does not limit the slave velocity.



Acceleration process for GEAR.MODE 0

Time	Description
t<t1	The master sends already signals to the slave drive, but the GEAR.MOVE command has not been triggered yet.
t = t1	A GEAR.MOVE command has been triggered.
t1<t<t2	The slave accelerates according to the GEAR.ACCMAX setting. The position, which is marked with solid lines, will be ignored.
t=t2	The slave has reached the master velocity and is from now on considered as synchronized. Synchronization means that the slave will not lose any more position counts coming from the master.
t>t2	The slave follows the master input signals.

Related Topics

Electronic Gearing (pg 117)

25.20.5 GEAR.MOVE

General Information	
Type	Command
Description	Starts the electronic gearing; active in opmode 2 (position) only.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	378	No	Command	No	M_01-03-00-000

Description

The command GEAR.MOVE starts the electronic gearing procedure according to the selected electronic gearing mode. The electronic gearing process can be stopped using the DRV.STOP (pg 556) command.

Related Topics

Electronic Gearing (pg 117)

25.20.6 GEAR.OUT

General Information	
Type	NV Parameter
Description	Sets the numerator of the electronic gearing ratio; active in opmode 2 (position) only.
Units	N/A
Range	-32,768 to +32,767
Default Value	1
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	380	No	16 bit	Yes	M_01-03-00-000

Description

This parameter is the numerator of the gear ratio for the electronic gearing mode. The gear ratio is used in order to increase/decrease the slave velocity. The slave velocity can be calculated by the following formula:

Slave velocity = Master velocity * GEAR.OUT/GEAR.IN (pg 643)

Make sure that the external master source has been set properly. Also, be certain to select a gear ratio such that the maximum electronic gearing velocity (GEAR.VMAX (pg 648)) will not be exceeded.

Master velocitymax * GEAR.OUT/GEAR.IN (pg 643) < GEAR.VMAX (pg 648)

Related Topics

Electronic Gearing (pg 117)

25.20.7 GEAR.VMAX

General Information	
Type	NV Parameter
Description	Reads the maximum allowed velocity value; active in opmode 2 (position) only.
Units	Depends on UNIT.ACCROTARY (pg 903) or UNIT.ACCLINEAR (pg 902) Rotary: rpm, rps, deg/s, Custom Units (pg 96)/s, rad/s Linear: counts/s, mm/s, µm/s, Custom Units (pg 96)/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 (pg 96)/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 (pg 761) mm/s 0.000 to 250,000.000*MOTOR.PITCH (pg 761) µm/sec 0.000 to 1,250.000 Custom Units (pg 96)/s
Default Value	Rotary: 3,000 rpm 50 rps 18,000.002 deg/s 250.000 Custom Units (pg 96)/s 314.159 rad/s Linear: 0.050 counts/s 50 mm/s 50,000.004MOTOR.PITCH (pg 761) µm/s 250.000 Custom Units (pg 96)/s
Data Type	Float
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	382	Yes	Low 32 bit word	No	M_01-03-00-000

Description

This parameter limits the maximum velocity of the slave drive.

Related Topics

Electronic Gearing (pg 117)

25.21 GUI Parameters

GUI parameters are used within WorkBenchfor data reporting and data storage.

25.21.1 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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.

25.21.2 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

25.21.3 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

25.21.4 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

25.21.5 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

25.21.6 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

25.21.7 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

25.21.8 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

25.21.9 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

25.21.10 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

25.21.11 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

The GUI uses this parameter to store data. Only the GUI can modify this data (not the user).

25.22 HOME Parameters

This section describes the HOME parameters.

25.22.1 HOME.ACC

General Information	
Type	R/W Parameter
Description	Sets homing acceleration; active in opmode 2 (position) only.
Units	Depends on UNIT.ACCROTARY (pg 903) or UNIT.ACCLINEAR (pg 902) Rotary: rps/s, rpm/s, deg/s^2, (Custom Units (pg 96))/s^2, rad/s^2 Linear: counts/s^2, mm/s^2, µm/s^2, (Custom Units (pg 96))/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 (pg 96))/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 (pg 761) to 833,333.333*MOTOR.PITCH (pg 761) mm/s^2 30.994*MOTOR.PITCH (pg 761) to 833,333,333.333*MOTOR.PITCH (pg 761) µm/s^2 0.155 to 4,166,666.667 (Custom Units (pg 96))/s^2
Default Value	Rotary: 166.669 rps/s 10,000.000 rpm/s 60,000.000 deg/s^2 833.333 (Custom Units (pg 96))/s^2 1,047.2 rad/s^2 Linear: 715,840,000.000 counts/s^2 166.714*MOTOR.PITCHMOTOR.PITCH (pg 761) mm/s^2 166,714.191*MOTOR.PITCHMOTOR.PITCH (pg 761) µm/s^2 833.571 (Custom Units (pg 96))/s^2
Data Type	Float
See Also	UNIT.ACCROTARY (pg 903), UNIT.ACCLINEAR (pg 902)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3502h/0 609Ah/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	384	Yes	64 bit	No	M_01-03-00-000

Description

This parameter determines the acceleration of the motor during the homing procedure.

Related Topics

Homing (pg 146)

25.22.2 HOME.AUTOMOVE

General Information	
Type	R/W Parameter
Description	Sets homing auto move flag.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Integer
See Also	HOME.MODE (pg 672)
Start Version	M_01-02-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	388	No	8 bit	No	M_01-03-00-000

Description

HOME.AUTOMOVE allows the drive to start a homing procedure after enabling the drive.

HOME.AUTOMOVE = 0: The drive is not allowed to start automatically a homing procedure after the enable command.

HOME.AUTOMOVE = 1: The drive automatically starts a homing procedure after the enable command.

When HOME.AUTOMOVE is set to 1, the drive continuously checks the following conditions:

1. Is the drive enabled (DRV.ACTIVE (pg 491) = 1)?
2. Is the drive in DRV.OPMODE (pg 549) = 2?
3. Has the command source been adjusted to 0 (DRV.CMDSOURCE (pg 497) = 0)?
4. Is no other motion currently active (see DRV.MOTIONSTAT (pg 541))?

As soon as all of the conditions above are true, the drive activates the homing procedure, which has been selected via the HOME.MODE (pg 672) setting. The automatic homing procedure is finished as soon as the homing procedure has been successfully triggered by the drive. The drive will not attempt to trigger any additional homing procedure until the drive has been power cycled.

HOME.AUTOMOVE is not functional in release M_01-03-00-000 for homing procedures which require an external index signal (HOME.MODE 3, 6, 10, and 11).

Related Topics:

Homing (pg 146)

25.22.3 HOME.DEC

General Information	
Type	R/W Parameter
Description	Sets homing deceleration; active in opmode 2 (position) only.
Units	Depends on UNIT.ACCROTARY (pg 903) or UNIT.ACCLINEAR (pg 902)UNIT.A-CCLINEAR (pg 902) Rotary: rps/s, rpm/s, deg/s^2, (Custom Units (pg 96))/s^2, rad/s^2 Linear: counts/s^2, mm/s^2, µm/s^2, (Custom Units (pg 96))/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 (pg 96))/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 (pg 761) to 833,333.333*MOTOR.PITCH (pg 761) mm/s^2 30.994*MOTOR.PITCH (pg 761) to 833,333,333.333*MOTOR.PITCH (pg 761) µm/s^2 0.155 to 4,166,666.667 (Custom Units (pg 96))/s^2
Default Value	Rotary: 166.669 rps/s 10,000.000 rpm/s 60,000.000 deg/s^2 833.333 (Custom Units (pg 96))/s^2 1,047.2 rad/s^2 Linear: 715,840,000.000 counts/s^2 166.714*MOTOR.PITCHMOTOR.PITCH (pg 761)mm/s^2 166,714.191MOTOR.PITCH (pg 761) µm/s^2 833.571 (Custom Units (pg 96))/s^2
Data Type	Float
See Also	UNIT.ACCROTARY (pg 903), UNIT.ACCLINEAR (pg 902)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3524h/0 609Ah/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	390	Yes	64 bit	No	M_01-03-00-000

Description

This parameter sets the deceleration of the motor during the homing procedure.

Related Topics:

Homing (pg 146)

25.22.4 HOME.DIR

General Information	
Type	NV Parameter
Description	Sets homing direction; active in opmode 2 (position) only.
Units	N/A
Range	0 to 1
Default Value	1
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	6098h	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	394	No	16 bit	No	M_01-03-00-000

Description

This parameter determines the direction in which the motor should start to move during a homing procedure.

0 = Movement in negative direction.

1 = Movement in positive direction.

Related Topics

Homing (pg 146)

25.22.5 HOME.DIST

General Information	
Type	R/W Parameter
Description	Sets homing distance; active in opmode 2 (position) only.
Units	Depends on UNIT.PROTARY (pg 908) or UNIT.PLINEAR (pg 906)UNIT.ACCLINEAR (pg 902) Rotary: counts, rad, deg, Custom Units (pg 96), 16-bit counts Linear: counts, mm, µm, Custom Units (pg 96), 16-bit counts
Range	N/A
Default Value	0
Data Type	Float
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	396	Yes	64 bit	Yess	M_01-03-00-000

Description

This parameter takes effect only after the homing procedure is complete (see the HOME.MODE (pg 672) description). HOME.DIST specifies an additional movement after the homing procedure is complete. The drive uses the homing acceleration, deceleration, and velocity parameters for this movement. This parameter can be used to let the motor move away from the home position by the value of HOME.DIST.

A value not equal to 0 triggers an additional movement of the selected homing distance after the general homing procedure. A value of 0 for HOME.DIST causes no additional movement.

Related Topics

Homing (pg 146)

25.22.6 HOME.FEEDRATE

General Information	
Type	R/W Parameter
Description	Sets homing velocity factor; active in opmode 2 (position) only.
Units	%
Range	0 to 100%
Default Value	50%
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	6099h/2	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	400	No	16 bit	No	M_01-03-00-000

Description

This parameter is used in order to reduce the velocity during the index search (index = zero-pulse of a feed-back device). This parameter determines the percentage of the homing velocity (HOME.V (pg 679)) that should be used during the index-search.

Related Topics

Homing (pg 146)

25.22.7 HOME.IPEAK

General Information	
Type	R/W Parameter
Description	Sets the current limit during homing procedure to a mechanical stop; active in opmode 2 (position) only.
Units	A
Range	± Drive peak current A
Default Value	[(1/120) * DRV.IPEAK (pg 535)] A
Data Type	Float
See Also	HOME.MODE (pg 672)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	35E2h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	402	Yes	64 bit	Yes	M_01-03-00-000

Description

This parameter sets the intermediate current limit during a homing procedure to a mechanical stop (HOME.MODE (pg 672) 8 and 9). The current-controller limit (IL.LIMITP (pg 710) and IL.LIMITN (pg 709)) is set to ±HOME.IPEAK while the homing procedures are active.

HOME.IPEAK is active as soon as the homing procedure starts and remains active until the home position is found. Previous current limit settings are re-activated before the motor covers the homing distance (HOME.DIST (pg 669) ≠ 0).

Related Topics

Homing (pg 146)Homing mode 8: Move Until Position Error Exceeded (pg 153)

25.22.8 HOME.MODE

General Information	
Type	R/W Parameter
Description	Selects the homing mode; active in opmode 2 (position) only.
Units	N/A
Range	0 to 10
Default Value	0
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	6098h	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	406	No	16 bit	No	M_01-03-00-000

Description

HOME.MODE specifies the homing procedure of the drive. The homing modes available in the drive are summarized in the following table; see Homing (pg 146) for a detailed discussion and examples for each mode:

Mode	Description
0	Home using current position
1	Find limit input
2	Find input limit then find zero angle
3	Find input limit then find index
4	Find home input, including hardware limit switches
5	Find home input then find zero angle, including hardware limit switches
6	Find home input then find index, including hardware limit switches.
7	Find zero angle
8	Move until position error exceeded
9	Move until position error exceeded, then find zero angle
10	Move until position error exceeded, then find index
11	Find index signal, without any precondition
12	Homing to a home-switch, including mechanical stop detection
13	Home using the feedback position

Related Topics

Homing (pg 146)

25.22.9 HOME.MOVE

General Information	
Type	Command
Description	Starts a homing procedure; active in opmode 2 (position) only.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Description

The HOME.MOVE command starts a homing procedure. The DRV.OPMODE (pg 549) must be set to 2 (closed position loop) and DRV.CMDSOURCE must be set to 0 (TCP/IP command).

Related Topics

Homing (pg 146)

25.22.10 HOME.P

General Information	
Type	R/W Parameter
Description	Sets home position; active in opmode 2 (position) only.
Units	Depends on UNIT.PROTARY (pg 908) or UNIT.PLINEAR (pg 906) Rotary: counts, rad, deg, Custom Units (pg 96), 16-bit counts Linear: counts, mm, µm, Custom Units (pg 96), 16-bit counts
Range	N/A
Default Value	0
Data Type	Float
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	607Ch/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	410	Yes	64 bit	Yes	M_01-03-00-000

Description

This parameter sets the home position. The command and actual position of the drive will be set to this value as soon as a homing event occurs. The homing events differ in each homing mode.

Related Topics

Homing (pg 146)

25.22.11 HOME.PERRTHRESH

General Information	
Type	R/W Parameter
Description	Sets the position lag threshold; active in opmode 2 (position) only.
Units	Depends on UNIT.PROTARY (pg 908) or UNIT.PLINEAR (pg 906) Rotary: counts, rad, deg, Custom Units (pg 96), 16-bit counts Linear: counts, mm, µm, Custom Units (pg 96), 16-bit counts
Range	N/A
Default Value	N/A
Data Type	Float
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3482h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	414	Yes	64 bit	Yes	M_01-03-00-000

Description

This parameter is used for the homing modes against a mechanical stop (HOME.MODE (pg 672) = 8 and 9). The absolute value of the following error (PL.ERR (pg 810)) is compared with HOME.PERRTHRESH in order to detect a mechanical stop.

Related Topics

Homing mode 8: Move Until Position Error Exceeded (pg 153)

25.22.12 HOME.REQUIRE

General Information	
Type	NV Parameter
Description	Defines if the axis must be homed before a motion task can be executed.
Units	N/A
Range	0 to 1
Default Value	1
Data Type	Integer
See Also	N/A
Start Version	M_01-03-00-005

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Description

This parameter defines whether or not the axis must be homed before a motion task can be executed.

- HOME.REQUIRE = 1: Homing must be complete ("Home Done" is true) before a motion task can execute.
- HOME.REQUIRE = 0: The axis does not need to be homed before a motion task can execute. When HOME.REQUIRE is set to 0, "Home Done" can be either true or false before a motion task can execute.

Related Topics

Homing (pg 146)
Motion Tasks (pg 157)

25.22.13 HOME.SET

General Information	
Type	Command
Description	Immediately sets the home position; active in opmode 2 (position) only.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	35F0h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	418	No	Command	No	M_01-03-00-000

Description

The HOME.SET command immediately homes the drive. The drive can be homed in an enabled or disabled state. Motion in the current mode of operation (DRV.OPMODE (pg 549)=0) or velocity mode of operation (DRV.OPMODE (pg 549)=1) is not affected by the HOME.SET command. Motion in the position mode of operation (DRV.OPMODE (pg 549)=2) is immediately aborted when the HOME.SET command is issued.

Related Topics

Homing (pg 146)

25.22.14 HOME.V

General Information	
Type	R/W Parameter
Description	Sets homing velocity; active in opmode 2 (position) only.
Units	Depends on UNIT.VROTARY (pg 910) or UNIT.VLINEAR (pg 909) Rotary: rpm, rps, deg/s, Custom Units (pg 96)/s, rad/s Linear: counts/s, mm/s, µm/s, Custom Units (pg 96)/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 (pg 96)/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 (pg 761) mm/s 0.000 to 250,000.000*MOTOR.PITCH (pg 761) µm/sec 0.000 to 1,250.000 Custom Units (pg 96)/s
Default Value	Rotary: 60 rpm 1 rps 359.999 deg/s 5 (Custom Units (pg 96))/s 6.283 rad/s Linear: 0.001 counts/s 1*MOTOR.PITCH (pg 761) mm/s 999.998*MOTOR.PITCH (pg 761) µm/sec 5.000 Custom Units (pg 96)/s
Data Type	Float
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	6099h/1	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	420	Yes	Low 32 bit word	No	M_01-03-00-000

Description

This parameter sets the velocity of the motor during the homing procedure.

Related Topics

Homing (pg 146)

25.23 HWLS Parameters

This section describes the HWLS parameters.

25.23.1 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 (pg 683)
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	422	No	8 bit	No	M_01-03-00-000

Description

HWLS.NEGSTATE reads the status of the negative HW limit switch as follows:

0 = Low

1 = High

Related Topics

25.23.2 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 (pg 682)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	424	No	8 bit	No	M_01-03-00-000

Description

HWLS.POSSTATE reads the status of the positive hardware limit switch as follows:

0 = Low

1 = High

Related Topics

25.24 IL Parameters

This section describes the IL parameters.

25.24.1 IL.BUSFF

General Information	
Type	R/O Parameter
Description	Displays the current feedforward value injected by the field-bus.
Units	Arms
Range	N/A
Default Value	N/A
Data Type	Float
See Also	IL.KBUSFF (pg 702)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	426	No	32 bit	Yes	M_01-03-00-000

Description

This parameter displays the current feedforward value injected by the fieldbus.

Related Topics

Current Loop (pg 134)

25.24.2 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 (pg 535)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	428	No	32 bit	Yes	M_01-03-00-000

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 (pg 709) and IL.LIMITP (pg 710).

Related Topics

13.3 Current Loop

25.24.3 IL.CMDACC

General Information	
Type	R/O Parameter
Description	Returns the commanded acceleration from the trajectory generator.
Units	Depends on UNIT.ACCLINEAR (pg 902) or UNIT.ACCROTARY (pg 903).
Range	+/- 64bit
Default Value	0
Data Type	Float
Start Version	M_01-08-03-000

Fieldbus Information

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.KACCFF).

Related Topics

IL.KACCFF (pg 701) | Tuning Guide (pg 212)

25.24.4 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 (pg 535), DRV.OPMODE (pg 549),DRV.CMDSOURCE (pg 497)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	430	No	32 bit	Yes	M_01-03-00-000

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 (pg 709) and IL.LIMITP (pg 710).

Related Topics

Current Loop (pg 134)

25.24.5 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3559h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1666	No	32-bits	No	M_01-06-03-000

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 (pg 710)), 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

13.3 Current Loop

25.24.6 IL.FB

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_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3558h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	432	No	32 bit	Yes	M_01-03-00-000

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 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

13.3 Current Loop

25.24.7 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 1
Default Value	0
Data Type	Integer
Start Version	M_01-08-00-000

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1730	No	8-bit	No

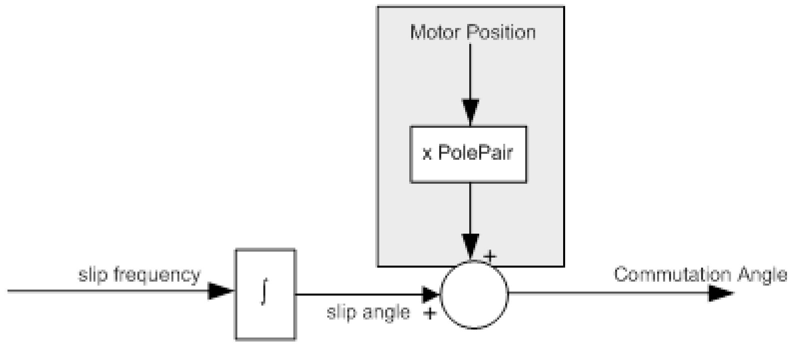
Description

This parameter selects the feedback source that current loop uses for commutation angle calculation.

0 = Primary feedback

1 = Secondary feedback

For induction motor closed loop control the commutation angle is set as shown in the following figure.



Related Topics

MOTOR.TYPE (pg 771)

25.24.8 IL.FF

General Information	
Type	R/O Parameter
Description	Displays the current loop overall feedforward value
Units	Arms
Range	N/A
Default Value	N/A
Data Type	Float
See Also	IL.KBUSFF (pg 702), IL.KVFF (pg 708), IL.OFFSET (pg 718), IL.FRICTION (pg 697), IL.KACCFF (pg 701)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	434	No	32 bit	No	M_01-03-00-000

Description

This parameter displays the current loop overall feedforward value.

Related Topics

Current Loop (pg 134)

25.24.9 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3420h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	436	No	32 bit	No	M_01-03-00-000

Description

IL.FOLDFTHRESH is the fault level of the current foldback algorithm. If IL.IFOLD (pg 698) 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

13.3 Current Loop

25.24.10 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 (pg 694), Foldback (pg 81)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3421h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	438	No	32 bit	Yes	M_01-03-00-000

Description

IL.FOLDFTHRESHU is the fault level of the current foldback algorithm. The value of IL.FOLDFTHRESH is the minimum of DRV.IPEAK (pg 535), MOTOR.IPEAK (pg 755), and IL.FOLDFTHRESHU.

Related Topics

13.3 Current Loop

25.24.11 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	355Ah/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	440	No	32 bit	Yes	M_01-03-00-000

Description

IL.FOLDWTHRESH is the warning level of the current foldback algorithm. When IL.IFOLD (pg 698) 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 (pg 694) value to zero.

Related Topics

13.3 Current Loop

25.24.12 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 (pg 710)
Default Value	0
Data Type	Float
See Also	IL.FF
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3422h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	442	No	32 bit	No	M_01-03-00-000

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

13.3 Current Loop

25.24.13 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3425h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	444	No	32 bit	No	M_01-03-00-000

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 (pg 710)), it becomes the active current limit.

Related Topics

13.3 Current Loop

25.24.14 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	446	No	32 bit	Yes	M_01-03-00-000

Description

This parameter displays the measured current in the u-winding of the motor.

Related Topics

Current Loop (pg 134)

25.24.15 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	448	No	32 bit	Yes	M_01-03-00-000

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

Current Loop (pg 134)

25.24.16 IL.KACCFF

General Information	
Type	R/W Parameter
Description	Sets current loop acceleration feedforward gain value
Units	mArms/(rad/s²)
Range	0.0 to 2.0 mArms/(rad/s²)
Default Value	0 mArms/(rad/s²)
Data Type	Float
See Also	IL.FF (pg 693)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3426h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	450	No	32 bit	Yes	M_01-03-00-000

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

13.3 Current Loop

25.24.17 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 (pg 693), IL.BUSFF (pg 686)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	452	No	32 bit	No	M_01-03-00-000

Description

This parameter scales the feedforward term added by the fieldbus to the current command. The nominal feedforward value can be multiplied by this gain value.

This parameter is only used in the position mode (DRV.OPMODE = 2).

Related Topics

Current Loop (pg 134)

25.24.18 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3598h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	454	No	16 bit	No	M_01-03-00-000

Description

IL.KP is used to modify the proportional gain of the PI-loop that controls the q-component of the current.

Related Topics

13.3 Current Loop

25.24.19 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 (pg 703)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3596h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	456	No	32 bit	No	M_01-03-00-000

Description

This parameter modifies the proportional gain of the PI-loop, which controls the d-component of the current.

Related Topics

13.3 Current Loop

25.24.20 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 (pg 706)IL.KPLOOKUPVALUES (pg 707)IL.KP (pg 703)
Start Version	M_01-04-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1226	No	16 bit	No

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 * DRV.IPEAK.

To determine the level of current that corresponds to a table index, use the following equation:

```
IL.CMD = (Table Index /157) * DRV.IPEAK
```

Related Topics

Current Loop (pg 134)

25.24.21 IL.KPLOOKUPVALUE

General Information	
Type	R/W Parameter
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 (pg 705)IL.KPLOOKUPVALUES (pg 707)IL.KP (pg 703)
Start Version	M_01-04-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1228	No	32 bit	No

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 * DRV.IPEAK. The value can range from 0% to 100% and determines what percentage of IL.KP will be applied to the current loop.

To determine what level of current corresponds to a table index, use the following equation:

```
IL.CMD = (Table Index/157) * DRV.IPEAK
```

Example

Assume:

DRV.IPEAK = 9 A

IL.KPLOOKUPINDEX = 100

IL.KPLOOKUPVALUE = 50

IL.KP = 240

When IL.CMD = 100/157 * 9 = 5.73 A, IL.KP will not be 240, but will be 50% * 240 = 120.

Related Topics

Current Loop (pg 134)

25.24.22 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 (pg 705), IL.KPLOOKUPVALUE (pg 706)IL.KP (pg 703)
Start Version	M_01-04-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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 (pg 134)

25.24.23 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 (pg 693)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	458	No	32 bit	Yes	M_01-03-00-000

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 (pg 549) = 2).

Related Topics

Current Loop (pg 134)

25.24.24 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 (pg 710)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	356Fh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	460	No	32 bit	Yes	M_01-03-00-000

Description

This parameter sets the negative user limit clamp value of the torqueproducing q-component current command (IL.CMD (pg 687)). The current command is additionally limited by the motor peak current setting (MOTOR.IPEAK (pg 755)) and by the present value of the foldback I²t peak motor current protection.

Related Topics

13.3 Current Loop

25.24.25 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 (pg 709)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	356Eh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	462	No	32 bit	Yes	M_01-03-00-000

Description

This parameter sets the positive user limit clamp value of the torque-producing q-component current command (IL.CMD (pg 687)). The current command is additionally limited by the motor peak current setting (MOTOR.IPEAK (pg 755)) and by the present value of the foldback I²t peak motor current protection.

Related Topics

13.3 Current Loop

25.24.26 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	464	No	32 bit	No	M_01-03-00-000

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

Current Loop (pg 134)

25.24.27 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	466	No	32 bit	No	M_01-03-00-000

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

Current Loop (pg 134)

25.24.28 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	468	No	32 bit	No	M_01-03-00-000

Description

IL.MFOLDT sets the time constant of the exponential drop (foldback) of the current towards motor continuous current.

Related Topics

Current Loop (pg 134)

25.24.29 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 (pg 715)
Start Version	M_01-04-01-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1184	No	16 bit	No

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%.

Related Topics

1 Motor I2t algorithm

25.24.30 IL.MI2TWTHRESH

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 (pg 714)
Start Version	M_01-04-01-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1668	No	8-bits	No	M_01-06-03-000

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.MI2TWTHRESH value. The warning n309 will be cleared as soon as IL.MI2T falls below the threshold.

Related Topics

Foldback (pg 81)

25.24.31 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	35A4h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	470	No	32 bit	No	M_01-03-00-000

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 (pg 710)) 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

13.3 Current Loop

25.24.32 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 (pg 714), IL.MI2TWTHRESH (pg 715)
Start Version	M_01-04-01-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1670	No	8-bits	No	M_01-06-03-000

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 mechanism is responsible for protecting the motor from overload.

Related Topics

Foldback (pg 81)

25.24.33 IL.OFFSET

General Information	
Type	RW Parameter
Description	A constant current command added to compensate for grav-ity.
Units	A
Range	[IL.LIMITN (pg 709) to IL.LIMITP (pg 710)
Default Value	0 A
Data Type	Float
See Also	IL.FF
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3423h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	472	No	32 bit	Yes	M_01-03-00-000

Description

This value is added to the overall current loop feedforward value.

Related Topics

13.3 Current Loop

25.24.34 IL.VCMD

General Information	
Type	R/O Parameter
Description	Sets the output of the q-component PI regulator.
Units	Vrms
Range	0 Vrms to bus voltage
Default Value	N/A
Data Type	Integer
See Also	IL.VDCMD
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	474	No	16 bit	Yes	M_01-03-00-000

Description

Sets the output of the current loop that controls the q-component of the current.

Related Topics

Current Loop (pg 134)

25.24.35 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 (pg 721)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	476	No	16 bit	Yes	M_01-03-00-000

Description

Reads the measured voltage on the u-winding of the motor.

Related Topics

Current Loop (pg 134)

25.24.36 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 (pg 720)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	478	No	16 bit	Yes	M_01-03-00-000

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

Current Loop (pg 134)

25.25 IP Parameters

This section describes the IP parameters.

25.25.1 IP.ADDRESS

General Information	
Type	NV Parameter
Description	Gets/Sets the IP address of the drive.
Units	N/A
Range	0.0.0.0 to 255.255.255.255
Default Value	0.0.0.0
Data Type	IP Address
See Also	
Start Version	M_01-04-05-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

This Parameter sets the IP address of the drive. If this parameter has not been set by the user, it will return 0.0.0.0.

By default, DHCP is active, and the drive will acquire an IP Address on its own. When the drive is in DHCP mode, IP.ADDRESS will return 0.0.0.0.

Notes:

- Even when the drive is in DHCP, the actual IP Address will not be returned using this command. The value the user has stored will be returned.
- IP.ADDRESS will only be used by the drive when IP.MODE = 1

If manually setting the IP.ADDRESS, the IP.SUBNET and IP.GATEWAYmust be set up. After the IP.RESET command is issued, the new IP settings will be active only if IP.MODE has been set to 1.

Recovering communications with a drive on an un-reachable IP address

Sometimes a drive may be configured for an IP Address, and the drive needs to be taken off-line, and bench tested, or otherwise used outside of its saved IP Settings. If IP.MODE has been set to 1 (using software defined static IP) the drive will boot up on an IP Address that may be unreachable with the host computer’s settings.

If the IP address prevents communication the IP settings can be reset to default by the following procedure:

- Set both rotary switches to 0
- Hold down button B1 (top-side of drive) for 5 seconds.

The display will flash 0.0.0.0 and then attempt to discover an address by DHCP. Without removing logic power from the drive, use Workbench to connect to the drive, reconfigure the IP address settings as desired, and store the values to non-volatile memory.

Related Topics

IP.GATEWAY | IP.RESET | IP.SUBNET | IP.MODE

25.25.2 IP.GATEWAY

General Information	
Type	NV Parameter
Description	Gets/Sets the gateway IP of the drive.
Units	N/A
Range	0.0.0.0 to 255.255.255.255
Default Value	0.0.0.0
Data Type	IP Address
See Also	Communicating with the Drive
Start Version	M_01-04-05-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

This Parameter sets the Gateway IP of the drive. This parameter determines what IP the drive can communicate with outside of its current subnet.

By default, DHCP is active, and the drive will acquire an IP Address on its own. When the drive is in DHCP mode, IP.GATEWAY will return 0.0.0.0.

Notes:

- When the drive is in DHCP, the actual IP GATEWAY will not be returned using this command. The value the user has stored will be returned.
- IP.GATEWAY will only be used by the drive when IP.MODE = 1

If manually setting the IP.ADDRESS (pg 723), the IP.SUBNET (pg 729) and IP.GATEWAY must be set up. After the IP.RESET (pg 727) command is issued, the new IP settings will be active only if IP.MODE (pg 725) has been set to 1.

Recovering communications with a drive on an un-reachable IP address

Sometimes a drive may be configured for an IP Address, and the drive needs to be taken off-line, and bench tested, or otherwise used outside of its saved IP Settings. If IP.MODE has been set to 1 (using software defined static IP) the drive will boot up on an IP Address that may be unreachable with the host computer's settings.

If the IP address prevents communication the IP settings can be reset to default by the following procedure:

- Set both rotary switches to 0
- Hold down button B1 (top-side of drive) for 5 seconds.

The display will flash 0.0.0.0 and then attempt to discover an address by DHCP. Without removing logic power from the drive, use Workbench to connect to the drive, reconfigure the IP address settings as desired, and store the values to non-volatile memory.

Related Topics

IP.ADDRESS (pg 723) | IP.RESET (pg 727) | IP.SUBNET (pg 729) | IP.MODE (pg 725)

25.25.3 IP.MODE

General Information	
Type	NV Parameter
Description	Sets method of acquiring IP Address.
Units	N/A
Range	0 to 2
Default Value	0
Data Type	Integer
See Also	N/A
Start Version	M_01-04-013-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1212	No	16 bit	No

Description

This command determines which method of acquiring an IP Address the drive will take.

NOTE Mode 0 and Mode 1 contain multiple methods of acquiring an IP Address. In these modes each method will be implemented in the order they are listed below until an IP Address is acquired

The drive will attempt to acquire a new IP Address as soon as the IP.RESET (pg 727) command is issued.

IP Mode	Mode of Acquiring IP Address
0	Rotary Switches, DHCP, Auto IP
1	IP.ADDRESS, IP.SUBNET, IP.GATEWAY
2	DHCP, Auto IP

Recovering communications with a drive on an un-reachable IP address

Sometimes a drive may be configured for an IP Address, and the drive needs to be taken off-line, and bench tested, or otherwise used outside of its saved IP Settings. If IP.MODE has been set to 1 (using software defined static IP), the drive will boot up on an IP Address that may be unreachable with the host computer's settings.

If the IP address prevents communication the IP settings can be reset to default by the following procedure:

- Set both rotary switches to 0
- Hold down button B1 (top-side of drive) for 5 seconds.

The display will flash 0.0.0.0 and then attempt to discover an address by DHCP. Without removing logic power from the drive, use Workbench to connect to the drive, reconfigure the IP address settings as desired, and store the values to non-volatile memory.

Related Topics

IP.ADDRESS (pg 723)

IP.GATEWAY (pg 724)

IP.RESET (pg 727)

IP.SUBNET (pg 729)

25.25.4 IP.RESET

General Information	
Type	Command
Description	Implements new IP settings.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
See Also	Communicating with the Drive
Start Version	M_01-04-05-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1672	No	Command	No	M_01-06-03-000

Description

When this command is issued a new IP will be acquired using IP.MODE (pg 725) to select what method is used.

Notes:

- When issuing this command, it is likely the connection to the drive will be severed, and a new connection will need to be made.
- IP.RESET will return an error if issued while the drive is enabled. IP.RESET is allowed when drive is disabled, or in dynamic braking mode.
- Ensure all values of IP.ADDRESS (pg 723), IP.SUBNET (pg 729), and IP.GATEWAY (pg 724) are configured if using IP.MODE 1

Recovering communications with a drive on an un-reachable IP address

Sometimes a drive may be configured for an IP Address, and the drive needs to be taken off-line, and bench tested, or otherwise used outside of its saved IP Settings. If IP.MODE has been set to 1 (using software defined static IP), the drive will boot up on an IP Address that may be unreachable with the host computer's settings.

If the IP address prevents communication the IP settings can be reset to default by the following procedure:

- Set both rotary switches to 0
- Hold down button B1 (top-side of drive) for 5 seconds.

The display will flash 0.0.0.0 and then attempt to discover an address by DHCP. Without removing logic power from the drive, use Workbench to connect to the drive, reconfigure the IP address settings as desired, and store the values to non-volatile memory.

Related Topics

IP.ADDRESS (pg 723) | IP.GATEWAY (pg 724) | IP.SUBNET (pg 729) | IP.MODE

25.25.5 IP.SUBNET

General Information	
Type	NV Parameter
Description	Gets/Sets the IP Subnet mask of the drive.
Units	N/A
Range	0.0.0.0 to 255.255.255.255
Default Value	0.0.0.0
Data Type	IP Address
See Also	
Start Version	M_01-04-05-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

This Parameter sets the IP Subnet mask of the drive. This parameter determines what IP addresses the drive will be allowed to communicate with.

By default, DHCP is active, and the drive will acquire an IP Address on its own. When the drive is in DHCP mode, IP.SUBNET will return 0.0.0.0.

Notes:

- When the drive is in DHCP, the actual IP Subnet mask will not be returned using this command. The value the user has stored will be returned.
- IP.SUBNET will only be used by the drive when IP.MODE = 1

If manually setting the IP.ADDRESS, the IP.SUBNET and IP.GATEWAYmust be set up. After the IP.RESET command is issued, the new IP settings will be active only if IP.MODE has been set to 1.

Recovering communications with a drive on an un-reachable IP address

Sometimes a drive may be configured for an IP Address, and the drive needs to be taken off-line, and bench tested, or otherwise used outside of its saved IP Settings. If IP.MODE has been set to 1 (using software defined static IP) the drive will boot up on an IP Address that may be unreachable with the host computer’s settings.

If the IP address prevents communication the IP settings can be reset to default by the following procedure:

- Set both rotary switches to 0
- Hold down button B1 (top-side of drive) for 5 seconds.

The display will flash 0.0.0.0 and then attempt to discover an address by DHCP. Without removing logic power from the drive, use Workbench to connect to the drive, reconfigure the IP address settings as desired, and store the values to non-volatile memory.

Related Topics

IP.ADDRESS | IP.GATEWAY | IP.RESET | 25.25.3 IP.MODE

25.26 LOAD-Parameter

This section describes the LOAD parameters.

25.26.1 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1214	No	32 bit	No

Description

LOAD.INERTIA sets the load inertia.

Related Topics

Motor (pg 64)

25.27 MODBUS Parameters

This section describes the MODBUS parameters.

25.27.1 MODBUS.CLRERRORS

General Information	
Type	Command
Description	Clears all errors stored in MOD-BUS.ERRORS.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
Start Version	M_01-08-03-001

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1758	No	Command	No

Description

Clears all errors stored in MODBUS.ERRORS. Error count is set to 0.

Related Topics

MODBUS.ERRORS (pg 735) | MODBUS.ERRORMODE (pg 734)

25.27.2 MODBUS.ERRORMODE

General Information	
Type	R/W
Description	Enable/disable Modbus error response messages.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Integer
Start Version	M_01-08-03-001

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1756	No	8 bit	No

Description

Enables or disables error response messages.

Value	Description
0	Send error response messages (default).
1	Do not send error response messages.

Related Topics

MODBUS.CLRERRORS (pg 733) | MODBUS.ERRORS (pg 735)

25.27.3 MODBUS.ERRORS

General Information	
Type	R/O
Description	Returns a list of up to 125 Modbus errors.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
Start Version	M_01-08-03-001

Description

Lists up to 125 Modbus errors. Each entry contains the Modbus address and the error code of the failed Modbus request.

If more than 125 errors occur, the oldest error is dropped and the new error is stored at the end of this list. All modbus errors are stored regardless of the state of the error mode (MODBUS.ERRORMODE).

Access Through Telnet

Telnet or the WorkBench terminal will display the entire list:

```
-->MODBUS.ERRORS
Error count: 2
[00384]   6 : Argument lower than minimum.
[00390] 109 : Invalid Register address.
```

Access Through Modbus

In Modbus the error list starts at register address 0x1102 (4354).

Modbus Address	Description
4354	Error count
4355	Error 1 : Register Address
4356	Error 1: Error Code
4357	Error 2 : Register Address
4358	Error 2 : Error Code
...	...
4603	Error 125 : Register Address
4604	Error 125 : Error Code

Related Topics

MODBUS.ERRORMODE (pg 734) | MODBUS.CLRERRORS (pg 733)

25.27.4 MODBUS.PIN

General Information	
Type	R/W
Description	Gets / Sets the Modbus User Units Input parameter.
Units	N/A
Range	1 to 4294967295
Default Value	1
Data Type	Integer
Start Version	M_01-04-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	974	No	32 bit	No

Description

MODBUS.PIN and MODBUS.POUT (pg 737) are used to apply user specified units to the feedback values retrieved over Modbus.

To use this parameter correctly, first the MODBUS.PSCALE (pg 738) must be known, this value determines the resolution per revolution of the motor for Modbus. Then the ratio of MODBUS.POUT (pg 737) /MODBUS.PIN is applied to convert counts/rev into User Units/rev.

Example

Use Modbus Scaling to return feedback in Radians

```
MODBUS.PSCALE = 16 (65536 counts/rev or pole pitch)
MODBUS.PIN = 5215189
MODBUS.POUT = 500000
```

If the motor is currently resting with a Modbus raw position 36,462 Counts (MODBUS.PSCALE (pg 738) is set to return 65,536 per rev) andthe user requests the position using PL.FB over Modbus, the position will be returned as:

```
36,462 * 500000 / 5215189 = 3495 (Radians * 1000)
```

Which equals 3.495 Radians

Related Topics

MODBUS.PSCALE (pg 738)

MODBUS.POUT (pg 737)

25.27.5 MODBUS.POUT

General Information	
Type	R/W
Description	Gets / Sets the Modbus User Units Output parameter.
Units	N/A
Range	1 to 4294967295
Default Value	1
Data Type	Integer
Start Version	M_01-04-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	976	No	32 bit	No

Description

MODBUS.PIN (pg 736) and MODBUS.POUT are used to apply User specified Units to the feedback values retrieved over Modbus.

To use this parameter correctly, first the MODBUS.PSCALE (pg 738) must be known. This value determines the resolution per revolution of the motor for Modbus. Then the ratio of MODBUS.POUT/MODBUS.PIN (pg 736) is applied to convert counts/rev into User Units/rev.

Example: Use Modbus Scaling to return feedback in Radians

```
MODBUS.PSCALE = 16 (65536 counts/rev or pole pitch)
MODBUS.PIN = 5215189
MODBUS.POUT = 500000
```

If the motor is currently resting with a Modbus raw position 36,462 Counts (MODBUS.PSCALE (pg 738) is set to return 65,536 per rev) andthe user requests the position using PL.FB over Modbus, the position will be returned as:

```
36,462 * 500000 / 5215189 = 3495 (Radians * 1000)
```

Which equals 3.495 Radians

Related Topics

MODBUS.PSCALE (pg 738)

MODBUS.PIN (pg 736)

Encoder Emulation (pg 77)

25.27.6 MODBUS.PSCALE

General Information	
Type	R/W
Description	Gets/Sets the Feedback Resolution (per rev) over Modbus.
Units	
Range	10 to 31
Default Value	20
Data Type	Integer
See Also	N/A
Start Version	M_01-04-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	978	No	16 bit	No

Description

This parameter determines the number of encoder counts per mechanical revolution reported over Modbus.

```
Modbus Resolution = 2^(MODBUS.PSCALE).
```

Notes:

- This scaling affects Position, Velocity and Acceleration when reading values over Modbus. This scaling term does not affect units over any other communication.
- Additionally, see MODBUS.PIN (pg 736) and MODBUS.POUT (pg 737), as these are applied on top of MODBUS.PSCALE to allow for user customizable units.

Related Topics

MODBUS.POUT (pg 737)

MODBUS.PIN (pg 736)

25.27.7 MODBUS.SCALING

General Information	
Type	NV Parameter
Description	Selects the scaling mode for Modbus values.
Units	N/A
Range	0 to 1
Default Value	1
Data Type	U8
Start Version	M_01-04-15-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1048	No	8 bit	No

Description

Modbus has its own scaling algorithms. For some simple HMI's, it is desirable to use Workbench units instead of these Modbus-specific units. The parameter MODBUS.SCALING allows users to disable Modbus scaling and enable WB (Telnet) scaling.

Setting	Description
0	Modbus uses same scaling units as Workbench (set by UNIT parameters)
1	Modbus uses the Modbus-specific scaling units (set by MODBUS parameters)

Related Topics

UNIT Parameters (pg 901)

25.27.8 MODBUS.UNITLABEL

General Information	
Type	R/W
Description	Labels the scaled resolution of a single motor turn.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	String
See Also	N/A
Start Version	M_01-04-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Description

The Modbus UnitLabel value is used to label the scaled resolution of a single motor turn.

Related Topics

Motor (pg 64)

25.28 MOTOR Parameters

This section describes the MOTOR parameters.

25.28.1 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3451h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	480	No	8 bit	No	M_01-03-00-000

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.

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 (pg 744)
- MOTOR.CTF0 (pg 748)
- MOTOR.ICONT (pg 749)
- MOTOR.INERTIA (pg 754)
- MOTOR.IPEAK (pg 755)
- MOTOR.KE (pg 756)
- MOTOR.KT (pg 757)
- MOTOR.LQLL (pg 758)
- MOTOR.NAME (pg 759)
- MOTOR.POLES (pg 762)
- MOTOR.R (pg 763)

If the motor's FB1.MEMVER > 0.01, AKD will also configure:

- MOTOR.PHASE (pg 760)
- MOTOR.TBRAKEAPP (pg 765)
- MOTOR.PHASE (pg 760)
- MOTOR.VRATED (pg 777)
- MOTOR.VOLTRATED (pg 776)

For firmware versions 01-09-00-000 and later, AKD will also configure:

- MOTOR.RTYPE (pg 764)
- MOTOR.TEMPFAULT (pg 769)

Related Topics

8.1 Motor

25.28.2 MOTOR.BRAKE

General Information	
Type	NV Parameter
Description	Sets the presence or absence of a motor brake.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Boolean
See Also	N/A
Start Version	M_01-00-00-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3587h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	482	No	8 bit	No	M_01-03-00-000

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.

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.AUTOSET=1.

Related Topics

8.1 Motor

25.28.3 MOTOR.BRAKEIMM

General Information	
Type	NV Parameter
Description	Brake Immediately: in the case of a drive disable, apply the brake in all situations.
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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1232	No	8 bit	No

Description

With the standard configuration, when the drive disables, the brake will not apply until velocity falls below CS.VTHRESH for CS.TO milliseconds. However, in some machines (such as a vertical axis) the brake should be applied immediately whenever the drive disables.

To ensure that the brake is applied immediately after any disable (due to fault, disable command, etc), set MOTOR.BRAKEIMM = 1.

Related Topics

Motor (pg 64) | CS.VTHRESH (pg 452) | CS.TO (pg 451) | MOTOR.TBRAKETO (pg 767) | DRV.DISTO (pg 511)

25.28.4 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3450h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	484	No	8 bit	No	M_01-03-00-000

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 A digital input mode is also used for the same purpose. The two mechanisms are independent.

NOTE The CANopen-object 0x345A sub 1/2 can be used to control the brake state. See CAN-open manual on kollmorgen.com. (Functionality starts with firmware 1.7.4.0)

Related Topics

Motor (pg 64)

25.28.5 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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 (pg 64)

25.28.6 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3454h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	486	No	32 bit	No	M_01-03-00-000

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 (pg 755) and MOTOR.ICONT (pg 749), determine the motor foldback parameters IL.MFOLDD (pg 711),IL.MFOLDT (pg 713), and IL.MFOLDR (pg 712).

Calculating MOTOR.CTF0

Given a motor coil/winding thermal time constant T in seconds, then:

```
MOTOR.CTF0 = 1/(2πT)
```

NOTE This parameter is automatically configured for Kollmorgen motors when MOTOR.A-UTOSET=1.

Related Topics

8.1 Motor

25.28.7 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

AKD BASIC Information	
Type	R/W

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	358Eh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	488	No	32 bit	No	M_01-03-00-000

Description

This parameter is used to configure the motor continuous current.

NOTE This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSET=1.

Related Topics

8.1 Motor

25.28.8 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	490	No	8 bit	No	M_01-03-00-000

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 varify OEM data integrity

Related Topics

Motor (pg 64)

25.28.9 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1732	No	32-bit	No

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 $\mathbf{I_d}$ is the value for IL.IMID, the preset rotor flux building current, $\mathbf{I_r}$ is the name plate phase current, **cosϕ** 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 (pg 64)

25.28.10 MOTOR.IMTR

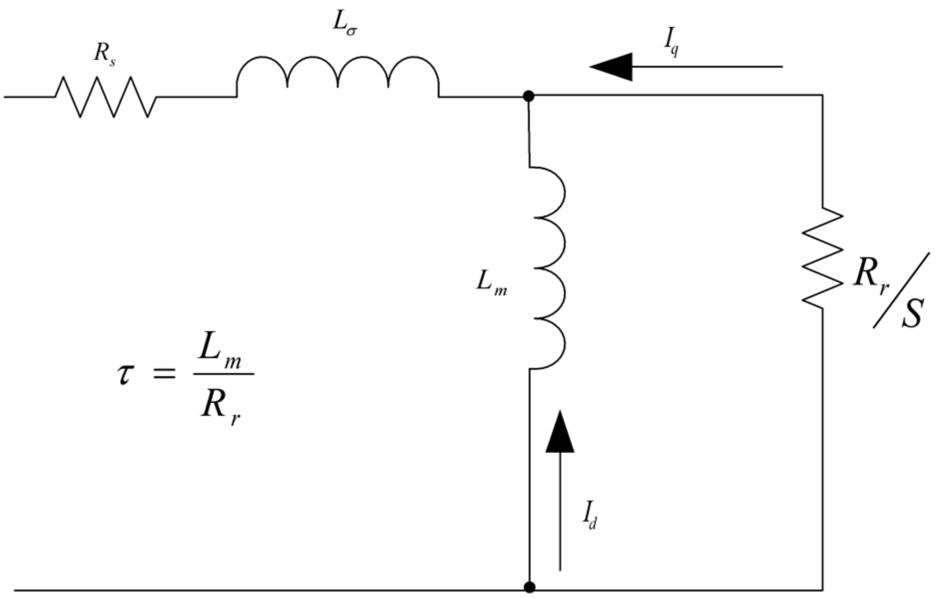
General Information	
Type	R/W
Description	Rotor time constant.
Units	Ms
Range	1 to 16,000
Default Value	100
Data Type	Integer
Start Version	M_01-08-00-000

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1728	No	16-bit	No

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 (pg 751)) 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 converted to ms for the

NOTE MOTOR.IMTR setting:
MOTOR.IMTR(ms) = τ*1000

Related Topics

Motor (pg 64)

25.28.11 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	35ABh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	492	No	32 bit	No	M_01-03-00-000

Description

This parameter sets the motor inertia.

NOTE This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSET=1.

Related Topics

8.1 Motor

25.28.12 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 (pg 710), IL.LIMITN (pg 709)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	358Fh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	494	No	32 bit	No	M_01-03-00-000

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 (pg 687)).

NOTE This parameter is automatically configured for Kollmorgen motors when MOTOR.A-UTOSET=1.

Related Topics

8.1 Motor

25.28.13 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1216	No	32 bit	No

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:

```
Coil Voltage = MOTOR.KE * VL.FB
```

Where:

VL.FB is in units of krpm for rotary motors and in units of m/s for linear motors

NOTE This parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSET=1.

Related Topics

Motor (pg 64)

25.28.14 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3593h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	496	No	32 bit	No	M_01-03-00-000

Description

This parameter is the torque constant of the motor in Nm/A.The value can be online checked according to the following equation:

```
Kt = 60 *√3 * Ui/( 2 * π* n)
```

Where:

Ui = induced voltage of the motor
n = actual rotor velocity

NOTE This parameter is automatically configured for Kollmorgen motors when MOTOR.A-UTOSET=1.

Related Topics

8.1 Motor

25.28.15 MOTOR.LQLL

General Information	
Type	NV Parameter
Description	Sets the line-to-line motor Lq.
Units	mH
Range	1 to 2^{32} H
Default Value	17.000 H
Data Type	Float
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3455h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	498	No	32 bit	No	M_01-03-00-000

Description

This parameter is used to configure the motor line-to-line inductance.

NOTE This parameter is automatically configured for Kollmorgen motors when MOTOR.A-UTOSET=1.

Related Topics

8.1 Motor

25.28.16 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

This parameter is used to set the motor name.

NOTE This parameter is automatically configured for Kollmorgen motors when MOTOR.A-UTOSET=1.

Related Topics

Motor (pg 64)

25.28.17 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	359Ch/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	500	No	16 bit	No	M_01-03-00-000

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.

Related Topics

8.1 Motor

25.28.18 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	502	No	32 bit	No	M_01-03-00-000

Description

This parameter defines the pole-to-pair pitch for the linear motor in micrometers.

Related Topics

Motor (pg 64)

25.28.19 MOTOR.POLES

General Information	
Type	NV Parameter
Description	Sets the number of motor poles.
Units	N/A
Range	0 to 128
Default Value	6
Data Type	Integer
See Also	FB1.POLES (pg 597)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	359Dh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	504	No	16 bit	No	M_01-03-00-000

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). 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.AUTOSET=1.

Related Topics

8.1 Motor

25.28.20 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3456h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	506	No	32 bit	No	M_01-03-00-000

Description

MOTOR.R sets the stator winding resistance phase-to-phase in ohms.

NOTE This parameter is automatically configured for Kollmorgen motors when MOTOR.A-UTOSET=1.

Related Topics

8.1 Motor

25.28.21 MOTOR.RTYPE

General Information	
Type	NV Parameter
Description	Defines the type of thermal resistor inside the motor.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Boolean
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	508	No	8 bit	No	M_01-03-00-000

Description

This parameter defines the type of thermal resistor used inside of the motor to measures motor temperature.

0 = PTC

1 = NTC

NOTE For firmware versions 01-09-00-000 and later, this parameter is automatically configured for Kollmorgen motors when MOTOR.AUTOSET = 1.

Related Topics

Motor (pg 64)

25.28.22 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	366Eh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	510	No	16 bit	No	M_01-03-00-000

Description

This parameter is used to configure the mechanical delay when applying the motor brake. MOTOR.TBRAKEAPP is a time delay that is applied when a brake exists and the drive is disabled at the end of a controlled stop. This delay lasts from the time that the brake is commanded to apply until the time that the drive is disabled.

This feature allows you to disable the drive and apply the brake on a vertical application without the load falling. Without this time delay, if you immediately disable the drive, then the load falls during the time needed for the brake to mechanically apply.

NOTE If MOTOR.AUTOSET = 1 and FB1.MEMVER > 0.01, this parameter will be automatically configured for Kollmorgen motors.

Related Topics

8.1 Motor

25.28.23 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	512	No	16 bit	No	M_01-03-00-000

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.AUTOSET = 1 and FB1.MEMVER > 0.01, this parameter will be automatically configured for Kollmorgen motors.

Related Topics

8.1 Motor

25.28.24 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 (pg 452), CS.TO (pg 451), DRV.DISTO (pg 511)
Start Version	01-05-07-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	990	No	32 bit	Yes

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 (pg 452). In some instances, such as a vertical axis, it may 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 (pg 452).

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.

25.28.25 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} Ω
Default Value	N/A
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3612h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	514	No	32 bit	No	M_01-03-00-000

Description

This parameter returns the motor temperature, which is represented as the resistance of the motor's PTC or NTC.

Related Topics

8.1 MotorMOTOR.RTYPE (pg 764)

25.28.26 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 (pg 768)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3586h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	516	No	32 bit	No	M_01-03-00-000

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.AUTOSET = 1.

Related Topics

8.1 Motor

25.28.27 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 (pg 768)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3453h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	518	No	32 bit	No	M_01-03-00-000

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

8.1 Motor

25.28.28 MOTOR.TYPE

General Information	
Type	NV Parameter
Description	Sets the motor type.
Units	N/A
Range	0, 1, 2 ,4
Default Value	0
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	520	No	8 bit	No	M_01-03-00-000

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

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.

Keyword	Permanent Magnet	Induction Motor v/f Open Loop	Induction Motor Closed Loop
MOTOR.NAME (pg 759)	Yes	Yes	Yes
MOTOR.TYPE	Yes	Yes	Yes
MOTOR.AUTOSET (pg 742)	Yes	No	Yes
MOTOR.IPEAK (pg 755)	Yes	Yes	Yes
MOTOR.ICONT (pg 749)	Yes	Yes	Yes
MOTOR.INERTIA (pg 754)	Yes	Yes	Yes
MOTOR.KT (pg 757)	Yes	No	No
MOTOR.LQLL (pg 758)	Yes	No	Yes
MOTOR.POLES (pg 762)	Yes	Yes	Yes
MOTOR.VMAX (pg 773)	Yes	Yes	Yes

Keyword	Permanent Magnet	Induction Motor v/f Open Loop	Induction Motor Closed Loop
MOTOR.R (pg 763)	Yes	Yes	Yes
MOTOR.VOLTMAX (pg 774)	Yes	Yes	Yes
MOTOR.PHASE (pg 760)	Yes	No	No
MOTOR.CTF0 (pg 748)	Yes	Yes	Yes
MOTOR.KE (pg 756)	Yes	No	No
MOTOR.IMTR (pg 752)	No	No	Yes
MOTOR.IMID (pg 751)	No	No	Yes
MOTOR.VOLTRATED (pg 776)	No	Yes	No
MOTOR.VRATED (pg 777)	No	Yes	No
MOTOR.VOLTMIN (pg 775)	No	Yes	No

Related Topics

Motor (pg 64)

25.28.29 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	35A3h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	522	No	16 bit	No	M_01-03-00-000

Description

This parameter is used to configure the maximum speed of the motor.

Related Topics

8.1 Motor

25.28.30 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3452h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	524	No	16 bit	No	M_01-03-00-000

Description

This parameter sets the maximum permissible motor voltage. For instance, if a motor that is rated for a 400 V supply is connected to the drive, then the MOTOR.VOLTMAX setting is 400. 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.

Related Topics

8.1 Motor

25.28.31 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 (pg 777), MOTOR.VOLTRATED (pg 776)
Start Version	

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3457h/3	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1674	No	16-bit	No	M_01-06-03-000

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

8.1 Motor

25.28.32 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 (pg 777), MOTOR.VOLTMIN (pg 775)
Start Version	M_01-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3457h/2	M_01-03-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1676	No	16-bit	No	M_01-06-03-000

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 (pg 64)

25.28.33 MOTOR.VRATED

General Information	
Type	NV Parameter
Description	Sets the motor rated velocity (not maximum velocity)
Units	Depends on or 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 (pg 776), MOTOR.VOLTMIN (pg 775)
Start Version	M_01-03-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3457h/1	M_01-03-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1678	Yes	64-bit	Yes	M_01-06-03-000

Description

This parameter configures the drive for the rated velocity of the induction motor as indicated on the name-plate. MOTOR.VRATED 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 (pg 64)

25.29 MT Parameters and Commands

This section describes the MT parameters and commands.

25.29.1 MT.ACC

General Information	
Type	R/W Parameter
Description	Specifies motion task acceleration; active in opmode 2 (position) only.
Units	Depends on UNIT.ACCROTARY (pg 903) or UNIT.ACCLINEAR (pg 902) Rotary: rps/s, rpm/s, deg/s², (Custom Units (pg 96))/s², rad/s² Linear: counts/s², mm/s², µm/s², (Custom Units (pg 96))/s²
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/s² 0.155 to 4,166,666.752 (Custom Units (pg 96))/s² 0.012 to 5,235,987.968 rad/s² Linear: 16,000.000 to 3,579,139,408,000.000 counts/s² 0.031*MOTOR.PITCH (pg 761) to 833,333.333*MOTOR.PITCH (pg 761) mm/s² 30.994*MOTOR.PITCH (pg 761) to 833,333,333.333*MOTOR.PITCH (pg 761)µm/s² 0.155 to 4,166,666.667 (Custom Units (pg 96))/s²
Default Value	Rotary: 166.669 rps/s 10,000.000 rpm/s 60,000.000 deg/s² 833.333 (Custom Units (pg 96))/s² 1,047.2 rad/s² Linear: 715,840,000.000 counts/s² 166.714*MOTOR.PITCHMOTOR.PITCH (pg 761) mm/s² 166,714.191*MOTOR.PITCHMOTOR.PITCH (pg 761) µm/s² 833.571 (Custom Units (pg 96))/s²
Data Type	Float
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	6083h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	526	Yes	64 bit	No	M_01-03-00-000

Description

MT.ACC specifies the motion task acceleration and is used by the MT.SET (pg 798) and MT.LOAD (pg 792) command. This parameter is a temporary value, since a motion task is only set after a MT.SET (pg 798) command. The motion task acceleration is further limited by the maximum allowed acceleration DRV.ACC

A value of 0 for MT.ACC should not be used when setting a motion task via MT.SET (pg 798) because this value causes a validity check of the MT.SET (pg 798) command to fail.

A value of 0 for MT.ACC after an MT.LOAD (pg 792) command displays an empty (not initialized) motion task.

Related Topics

Motion Tasks (pg 157)

25.29.2 MT.CLEAR

General Information	
Type	Command
Description	Clears motion tasks from the drive; active in opmode 2 (position) only.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC (pg 780), MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	530	No	16 bit	Yes	M_01-03-00-000

Description

MT.CLEAR clears a motion task from the drive. This command needs one argument in order to clear a motion task. A motion task consists of the following parameters: MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC (pg 780),MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799)

A value of –1 clears all motion tasks in the drive (MT.CLEAR –1).

Example

MT.CLEAR 5: Clear motion task number 5.

After performing a command such as MT.PARAMS 5, the drive displays the following:

```
5   0.000 Counts   0.000 rpm   0   0.000 rpm/s   0.000 rpm/s   0
0   0 ms
```

A value of 0 for velocity, acceleration, or deceleration displays motion task as uninitialized.

Related Topics

Motion Tasks (pg 157)

25.29.3 MT.CNTL

General Information	
Type	R/W Parameter
Description	Sets motion task control word; active in opmode 2 (position) only.
Units	N/A
Range	0 to 4,294,967,295
Default Value	0
Data Type	Integer
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.ACC (pg 780)MT.V (pg 805), MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794) MT.MTNEXT (pg 794), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	35AFh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	532	No	32 bit	No	M_01-03-00-000

Description

MT.CNTL specifies the motion task control word, which is used by the MT.SET (pg 798) and MT.LOAD (pg 792) commands. The control word describes the behavior of the motion task. This parameter is a temporary value, since a motion task is only set after an MT.SET (pg 798) command is issued.

Since this parameter is read bitwise, it can combine multiple functions into a single word. The meaning of each bit is described in the tables below.

Table 1: Motion Task (MT) Bit Descriptions

Bit	Meaning	Description
0	0x00001	See Table 2: MT Type.
1	0x00002	
2	0x00004	
3	0x00008	
4	0x00010	If this bit is 0, then the next MT is not executed. If this bit is 1, then the next MT is executed.

Bit	Meaning	Description
5	0x00020	See Table 3: Next MT Start Type.
6	0x00040	
7	0x00080	
8	0x00100	
9	0x00200	
10	0x00400	See Table 4: MT Acceleration Type.
11	0x00800	
12	0x01000	Activates the override functionality for a trapezoidal MT. If this bit is 1, a motion task with override functionality must be activated (see bit 5).
13	0x02000	If this bit is 0, then an attempt to trigger any new motion task will be accepted while this motion task is currently running. If this bit is 1, then an attempt to trigger any new motion task will be denied while this motion task is currently running.
14	0x04000	If this bit is set, the motion task that is supposed to be started cannot be started from velocity 0. The motion can be started if a motion task already running will be interrupted.
15	0x08000	Reserved.
16	0x10000	The motion task target velocity will be taken from an external source such as an analog input signal (see AIN.MODE (pg 371) for further details).

Table 2: MT Type

Bits 3, 2, 1, 0	Description
0000	Absolute. The target position is defined by the MT.P value.
1000	Reserved.
0001	Relative to Command Position. The target position is defined as: Target position = PL.CMD (pg 809) + MT.P (pg 796)
0011	Relative to Previous Target Position. The target position is defined as: Target position = Target position of the last motion task + MT.P
0101	Reserved.
0111	Relative to Feedback Position. The target position is defined as: Target position = PL.FB (pg 817) + MT.P (pg 796)

Table 3: Next MT Start Type

Bits 9, 8, 7, 6, 5	Description
00000	Switches over to next MT after stopping. After an MT ends, the next MT starts immediately.
00001	Switches over to next MT after stopping and delay. After an MT ends, the MT following time (MT.TNEXT (pg 799)elapse in order to start the next MT.
00010	Switches over to next MT after stopping and external event. After an MT ends, an external event (such as a high digital input) must occur in order to start the next MT.
00011	Switches over to next MT after stopping, delay, and external event. After an MT ends, the MT.TNEXT (pg 799)must elapse and an external event (such as a high digital input) must occur in order to start the next MT.

Bits 9, 8, 7, 6, 5	Description
00111	Switches over to next MT after stopping, then delay or external event. After an MT ends, the MT.TNEXT (pg 799) must elapse or an external event (such as a high digital input) must occur in order to start the next MT.
10000	Switches over to the next MT at present MT speed (change on the fly). After reaching the target position of an MT, the next MT starts. The drive then accelerates with the adjusted acceleration ramp of this next MT to the target velocity of this next MT. The MT.TNEXT (pg 799) setting is ignored.
11000	Switches over to the next MT at next MT speed (change on the fly).When the target position of an MT is reached, the drive has already accelerated with the acceleration ramp of the next MT to the target velocity of the next MT. Thus, the drive begins the next MT at the next MT target velocity. The MT.TNEXT (pg 799) setting is ignored if adjusted.

Table 4: MT Acceleration Type

Bits 11, 10	Description
00	Trapezoidal acceleration and deceleration.
01	1:1 motion profile table motion task. The drive follows the customer motion profile table without inserting a constant velocity phase between the acceleration and deceleration process. This setting allows the usage of nonsymmetric velocity profiles. The MT.TNUM parameter defines which table to use for the 1:1 profile handling.
11	Standard motion profile table motion task. The drive accelerates according to the shape of the motion profile table by stepping through the first half of the customer table. Then the drive inserts a constant velocity phase until the brake point is reached. Finally, the drive decelerates by stepping through the second half of the customer profile table. The MT.TNUM parameter defines which table to use for the 1:1 profile handling. This mode allows also a change on the fly between motion tasks (see Table 3 above). See "AKD Customer Profile Application Note" on the Kollmorgen web site (www.kollmorgen.com) for additional details.

Related Topics

Motion Tasks (pg 157)

25.29.4 MT.CONTINUE

General Information	
Type	Command
Description	Continues a stopped motion task; active in opmode 2 (position) only.
Units	N/A
Range	N/A
Default Value	0
Data Type	N/A
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC (pg 780) MT.V (pg 805), MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794) MT.MTNEXT (pg 794), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	534	No	Command	No	M_01-03-00-000

Description

MT.CONTINUE continues a motion task that has been stopped by the DRV.STOP command.

Related Topics

Motion Tasks (pg 157)

25.29.5 MT.DEC

General Information	
Type	R/W Parameter
Description	Sets motion task deceleration; active in opmode 2 (position) only.
Units	Depends on UNIT.ACCROTARY (pg 903) or UNIT.ACCLINEAR (pg 902) Rotary: rps/s, rpm/s, deg/s², (Custom Units (pg 96))/s², rad/s² Linear: counts/s², mm/s², µm/s², (Custom Units (pg 96))/s²
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/s² 0.155 to 4,166,666.752 (Custom Units (pg 96))/s² 0.012 to 5,235,987.968 rad/s² Linear: 16,000.000 to 3,579,139,408,000.000 counts/s² 0.031*MOTOR.PITCH (pg 761) to 833333.333*MOTOR.PITCH (pg 761) mm/s² 30.994*MOTOR.PITCH (pg 761) to 833333333.333*MOTOR.PITCH (pg 761) µm/s² 0.155 to 4166666.667 (Custom Units (pg 96))/s²
Default Value	Rotary: 166.669 rps/s 10,000.000 rpm/s 60,000.000 deg/s² 833.333 (Custom Units (pg 96))/s² 1,047.2 rad/s² Linear: 715,840,000.000 counts/s² 166.714*MOTOR.PITCHMOTOR.PITCH (pg 761) mm/s² 166,714.191*MOTOR.PITCHMOTOR.PITCH (pg 761) µm/s² 833.571 (Custom Units (pg 96))/s²
Data Type	Float
See Also	MT.ACC (pg 780), MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	6084h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	536	Yes	64 bit	No	M_01-03-00-000

Description

MT.DEC specifies the motion task deceleration and is used by the MT.SET (pg 798) and MT.LOAD (pg 792) commands. This parameter is a temporary value, since a motion task is only set after an MT.SET command is issued. The motion task deceleration is further limited by the maximum allowed acceleration, DRV.DEC.

A value of 0 for MT.DEC should not be used when setting a motion task via MT.SET because this value causes a validity check of the MT.SET command to fail.

A value of 0 for MT.DEC after an MT.LOAD (pg 792) command displays an empty (not initialized) motion task.

Related Topics

Motion Tasks (pg 157)

25.29.6 MT.EMERGMT

General Information	
Type	R/W Parameter
Description	Selects a motion task to be triggered after an emergency stop procedure; active in opmode 2 (position) only.
Units	N/A
Range	1 to 128
Default Value	0
Data Type	N/A
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	540	No	16 bit	Yes	M_01-03-00-000

Description

MT.EMERGMT selects the motion task to be triggered after an emergency stop procedure.

A value of –1 shows that no motion task must be started after a ramp-down procedure in a closed position loop mode of operation.

Related Topics

Motion Tasks (pg 157)

25.29.7 MT.HOMEREQUIRE

General Information	
Type	NV Parameter
Description	Removed in 01-04-00-000.
Units	N/A
Range	0 to 1
Default Value	1
Data Type	Integer
See Also	N/A
Start Version	M_01-03-00-005 (removed in 01-04-00-000)

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1682	No	8-bit	No	M_01-06-03-000

Description

Removed in 01-04-00-000.

Related Topics

Motion Tasks (pg 157)

25.29.8 MT.LIST

General Information	
Type	Command
Description	Lists all initialized motion tasks in the drive; active in opmode 2 (position) only.
Units	N/A
Range	0
Default Value	N/A
Data Type	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
See Also	M_01-00-00-000
Start Version	N/A

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Description

MT.LIST reads every initialized motion task from the drive. A motion task consists of the following parameters: MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), and MT.TNEXT (pg 799).

A motion task is considered as initialized as soon as MT.V (pg 805), MT.ACC, and MT.DEC (pg 787) of that specific motion task have values not equal to 0.

Related Topics

Motion Tasks (pg 157)

25.29.9 MT.LOAD

General Information	
Type	Command
Description	Reads/loads a motion task number from the drive; active in opmode 2 (position) only.
Units	N/A
Range	N/A
Default Value	0
Data Type	N/A
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	542	No	Command	No	M_01-03-00-000

Description

MT.LOAD (pg 792) reads out a motion task number MT.NUM (pg 795) from the drive. A motion task consists of the following parameters: MT.NUM, MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799). These parameters belong to the motion task number MT.NUM and are refreshed by MT.LOAD.

Related Topics

Motion Tasks (pg 157)

25.29.10 MT.MOVE

General Information	
Type	Command
Description	Starts a motion task; active in opmode 2 (position) only.
Units	N/A
Range	N/A
Default Value	0
Data Type	N/A
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	544	No	Command	No	M_01-03-00-000

Description

MT.MOVE starts a motion task. This command needs one argument in order to start a motion task. The drive must be homed, otherwise the motion task will not start (see also HOME commands).

Example

MT.MOVE 3 -> Start motion task number 3.

Related Topics

Motion Tasks (pg 157)

Digital Inputs and Outputs (pg 100) in Digital Inputs and Outputs (pg 100) (see also **Modes 3: Motion Task Select Bit** and **Mode 4: Motion Task Start Selected** in this topic)

25.29.11 MT.MTNEXT

General Information	
Type	R/W Parameter
Description	Specifies following motion task number; active in opmode 2 (position) only.
Units	N/A
Range	0 to 128
Default Value	0
Data Type	Integer
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	35BCh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	546	No	8 bit	No	M_01-03-00-000

Description

MT.MTNEXT specifies the number of the following motion task and is used by the MT.SET (pg 798) and MT.LOAD (pg 792) command. This parameter is a temporary value. A motion task is only set after an MT.SET (pg 798) command.

The motion task control word can be selected so that a following motion task is executed after a first motion task. This parameter displays which motion task should be started after the first motion task.

Related Topics

Motion Tasks (pg 157)

25.29.12 MT.NUM

General Information	
Type	R/W Parameter
Description	Sets the motion task number; active in opmode 2 (position) only.
Units	N/A
Range	0 to 128
Default Value	0
Data Type	Integer
See Also	MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	365Bh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	548	No	8 bit	No	M_01-03-00-000

Description

MT.NUM (pg 795) specifies the motion task number, which is used by the MT.SET (pg 798) and MT.LOAD (pg 792) commands. This parameter is a temporary value. A motion task is only set after an MT.SET (pg 798) command is issued.

Related Topics

Motion Tasks (pg 157)

25.29.13 MT.P

General Information	
Type	R/W Parameter
Description	Sets the motion task position; active in opmode 2 (position) only.
Units	Depends on UNIT.PROTARY (pg 908) or UNIT.PLINEAR
Range	N/A
Default Value	0
Data Type	Float
See Also	MT.NUM (pg 795), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	607Ah/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	550	Yes	64 bit	Yes	M_01-03-00-000

Description

MT.P specifies the motion task position, which is used by the MT.SET (pg 798) and MT.LOAD (pg 792) command. Depending on the motion task control word (MT.CNTL), the MT.P command can either be the target position of the motion task or a relative distance. This parameter is a temporary value. A motion task is only set after an MT.SET (pg 798) command.

Related Topics

Motion Tasks (pg 157)

25.29.14 MT.PARAMS

General Information	
Type	Command
Description	Shows a motion task; active in opmode 2 (position) only.
Units	N/A
Range	N/A
Default Value	0
Data Type	N/A
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Description

MT.PARAMS displays a motion task. This command needs one argument in order to show a motion task. If you enter MT.PARAMS without an argument, the drive returns the current or last active motion task.

Example

MT.PARAMS 5

The drive responds as follows:

```
7   5222.000 Counts   135.000 rpm   1   550.746 rpm/s   654.458
rpm/s   0   0   0 ms
```

Related Topics

Motion Tasks (pg 157)

25.29.15 MT.SET

General Information	
Type	Command
Description	Sets the motion task in the drive; active in opmode 2 (position) only.
Units	N/A
Range	N/A
Default Value	0
Data Type	N/A
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3431h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	554	No	Command	No	M_01-03-00-000

Description

MT.SET sends a motion task to the drive. A motion task consists of the following parameters: MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), and MT.TNEXT (pg 799).

The motion task number (MT.NUM (pg 795)) with the parameters above is sent to the drive only after the MT.SET command.

Related Topics

Motion Tasks (pg 157)

25.29.16 MT.TNEXT

General Information	
Type	R/W Parameter
Description	Specifies following motion task time; active in opmode 2 (position) only.
Units	ms
Range	0 to 65,535 ms
Default Value	0 ms
Data Type	Integer
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	35BDh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	556	No	16 bit	No	M_01-03-00-000

Description

MT.TNEXTspecifies the time that must elapse before starting a following motion task. This value is used by the MT.SET (pg 798) and MT.LOAD (pg 792) command. This parameter is a temporary value. A motion task is only set after an MT.SET (pg 798) command.

The motion task control word can be selected so that a following motion task is executed after a first motion task and this additional delay time.

Related Topics

Motion Tasks (pg 157)

25.29.17 MT.TNUM

General Information	
Type	R/W Parameter
Description	Sets the motion task customer table number; active in opmode 2 (position) only.
Units	N/A
Range	0 to 7
Default Value	0
Data Type	Integer
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.V (pg 805), MT.CNTL (pg 783), MT.ACC, MT.DEC (pg 787), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	558	No	8 bit	No	M_01-03-00-000

Description

MT.TNUM (pg 800) specifies the customer profile table and is used by the MT.SET (pg 798) and MT.LOAD (pg 792) command. This parameter is a temporary value. A motion task is only set after an MT.SET (pg 798) command.

The drive can have up to eight customer specific profile tables. The drive performs an S-curve acceleration with these profile tables. The shapes of these tables have an impact on the shape of the motion task acceleration and deceleration. The motion task control word specifies if a customer profile table is used or not.

This parameter has no impact when a trapezoidal motion task acceleration and deceleration profile is selected (see Table 4: MT Acceleration Type (pg 785)in the MT.CNTL description).

Related Topics

Motion Tasks (pg 157)

25.29.18 MT.TNVSAVE

General Information	
Type	Command
Description	Saves the motion profile tables to the nonvolatile memory.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
See Also	MT.TDWNLDS, MT.TDWNLDV
Start Version	M_01-04-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

Description

This command saves all motion profile tables, which are used for motion tasking, to the nonvolatile memory. The nonvolatile memory for these tables can be cleared by triggering this command while no motion profile table is available in the volatile memory (empty array in the volatile memory). The nonvolatile memory for these tables can be deleted as follows:

```
-->MT.TDWNLDS 1
-->MT.TDWNLDS 2
-->MT.TNVSAVE
```

The first two commands are needed in order to delete all motion profile tables in the volatile memory. The following MT.TNVSAVE command detects that there is no data available in the volatile memory and therefore deletes the nonvolatile memory sector.

WorkBench also uses this command for the firmware download procedure.

25.29.19 MT.TPOSWND

General Information	
Type	R/W Parameter
Description	Sets the motion task target position window; active in opmode 2 (position) only.
Units	Depends on UNIT.PROTARY (pg 908) or UNIT.PLINEAR (pg 906) Rotary: counts, rad, deg, Custom Units (pg 96), 16-bit counts Linear: counts, mm, µm, Custom Units (pg 96), 16-bit counts
Range	N/A
Default Value	0.5 rev
Data Type	Float
See Also	DRV.MOTIONSTAT (pg 541)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	35C6h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	560	Yes	64 bit	Yes	M_01-03-00-000

Description

Within DRV.MOTIONSTAT, MT.TPOSWND is used to indicate that the target position of a motion task has been reached. DRV.MOTIONSTAT displays a "Target Position Reached" bit as soon as the following statement becomes true:

abs(actual_position – target_position) < MT.TPOSWND

NOTE DOUT3.MODE and DOUT17.MODE (MT in Position) are almost identical. Mode 17 will trigger as soon as the load is in the position window, whereas Mode 3 will wait until the trajectory is complete before monitoring the window. Mode 17 may signal faster because of this, and can also potentially bounce out of the window temporarily.

Related Topics

Motion Tasks (pg 157)

25.29.20 MT.TVELWND

General Information	
Type	R/W Parameter
Description	Sets the motion task target velocity window; active in opmode 2 (position) only.
Units	Depends on UNIT.VROTARY (pg 910) or UNIT.VLINEAR (pg 909) Rotary: rpm, rps, deg/s, Custom Units (pg 96)/s, rad/s Linear: Counts/s, mm/s, µm/s, Custom Units (pg 96)/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 (pg 96)/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 (pg 761) mm/s 0.000 to 250,000.000*MOTOR.PITCH (pg 761) µm/sec 0.000 to 1,250.000 Custom Units (pg 96)/s
Default Value	Rotary: 60.000 rpm 1.000 rps 359.999 deg/s 5.000 Custom Units (pg 96)/s 6.283 rad/s Linear: 0.001 counts/s 1.000*MOTOR.PITCH (pg 761) mm/s 999.998*MOTOR.PITCH (pg 761) µm/sec 5.000 Custom Units (pg 96)/s
Data Type	Float
See Also	DRV.MOTIONSTAT
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3856h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	564	No	32 bit	No	M_01-03-00-000

Description

Within DRV.MOTIONSTAT (pg 541), MT.TVELWND is used to indicate that the target velocity of a motion task has been reached. DRV.MOTIONSTAT (pg 541) displays a "Target Velocity Reached" bit as soon as the following statement becomes true:

(target velocity – MT.TVELWND) < actual velocity < (target velocity + MT)

Related Topics

Motion Tasks (pg 157)

25.29.21 MT.V

General Information	
Type	R/W Parameter
Description	Sets the motion task velocity; active in opmode 2 (position) only.
Units	Depends on UNIT.VROTARY (pg 910) or UNIT.VLINEAR (pg 909) Rotary: rpm, rps, deg/s, Custom Units (pg 96)/s, rad/s Linear: counts/s, mm/s, µm/s, Custom Units (pg 96)/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 (pg 96)/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 (pg 761) mm/s 0.000 to 250,000.000*MOTOR.PITCH (pg 761) µm/sec 0.000 to 1,250.000 Custom Units (pg 96)/s
Default Value	0
Data Type	Float
See Also	MT.NUM (pg 795), MT.P (pg 796), MT.CNTL (pg 783), MT.ACC (pg 780), MT.DEC (pg 787), MT.TNUM (pg 800), MT.MTNEXT (pg 794), MT.TNEXT (pg 799), MT.SET (pg 798), MT.LOAD (pg 792)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	6081h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	566	Yes	Low 32 bit word	No	M_01-03-00-000

Description

MT.V specifies the motion task velocity, which is used by the MT.SET (pg 798) and MT.LOAD (pg 792) command. This parameter is a temporary value. A motion task is only set after an MT.SET (pg 798) command. The motion task velocity is furthermore limited by VL.LIMITP or VL.LIMITN depending on the direction of the motion task.

A value of 0 should not be used when setting a motion task via MT.SET (pg 798) because this value causes a validity check of the MT.SET (pg 798) command to fail.

A value of 0 after an MT.LOAD (pg 792) command displays an empty (not initialized) motion task.

Related Topics

Motion Tasks (pg 157)

25.29.22 MT.VCMD

General Information	
Type	R/O Parameter
Description	Reads the derivative of PL.CMD; active in opmode 2 (position) only.
Units	Depends on UNIT.VROTARY (pg 910) or UNIT.VLINEAR (pg 909)
Range	N/A
Default Value	N/A
Data Type	Float
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	568	Yes	Low 32 bit word	Yes	M_01-03-00-000

Description

MT.VCMD returns the derivative of the position loop trajectory (PL.CMD), which is therefore a velocity. MT.VCMD is updated while the drive is in DRV.OPMODE 2 and is processing the following motion types:

- Motion tasking
- Homing
- Electronic gearing
- Service motion
- External trajectory coming from a fieldbus
- External trajectory calculated from an analog input signal

Related Topics

Motion Tasks (pg 157)

25.30 PL Parameters

This section describes the PL parameters.

25.30.1 PL.CMD

General Information	
Type	R/O Parameter
Description	Reads the position command directly from the entry to the position loop.
Units	Depends on or 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	Float
See Also	PL.FB (pg 817)
Start Version	M_01-00-00-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	570	Yes	64 bit	No	M_01-03-00-000

Description

PL.CMD reads the position command as it is received in the position loop entry.

Related Topics

PL.ERR (pg 810) | PL.ERRFTHRESH (pg 811) | PL.ERRMODE (pg 813) | PL.ERRWTHRESH (pg 815)

13.5 Position Loop

25.30.2 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 (pg 817)
Start Version	M_01-00-00-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	35C5h/0 60F4h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	574	Yes	64 bit	No	M_01-03-00-000

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 (pg 817)) and the commanded position of the drive (PL.CMD (pg 809)). If the drive is not in the position operating mode (DRV.OPMODE (pg 549) = 2), then the PL.ERR value is not generated by the drive and this parameter is read as 0.

AKD BASIC Notes

NOTE When you enable the position error interrupt (by setting INTR.PL.ERR=1), the Position Error fault is disabled. In situations where it would have occurred, a position error interrupt is generated instead.

Related Topics

PL.ERRFTHRESH (pg 811) | PL.ERRMODE (pg 813) | PL.ERRWTHRESH (pg 815)

13.5 Position Loop

25.30.3 PL.ERRFTHRESH

General Information	
Type	NV Parameter
Description	Sets the maximum position error.
Units	Depends on or 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 (pg 761) mm 0.000 to 1,192,878,014.464*MOTOR.PITCH (pg 761) µm 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 (pg 761)mm 10,000.000*MOTOR.PITCH µm 50.000 (custom units) 655,360.000 16-bit counts
Data Type	Float
See Also	PL.ERR
Start Version	M_01-00-00-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	35C7h/0 6065h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	580	Yes	64 bit	No	M_01-03-00-000

Description

This parameter sets the maximum position error. If the position error PL.ERR (pg 810) 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 is the position error is larger than 1000 degrees, the drive will generate a fault.

UNIT.PROTARY 2

PL.ERRFTHRESH 1000

Related Topics

PL.ERR (pg 810) | PL.ERRMODE (pg 813) | PL.ERRWTHRESH (pg 815)

13.5 Position Loop

25.30.4 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 (pg 810), PL.ERRFTHRESH (pg 811), PL.ERRWTHRESH (pg 815)
Start Version	M_01-02-09-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	578	No	8 bit	No	M_01-03-00-000

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:

<error> = abs(PL.ERR - [(VL.CMD – 1*VL.FF) / PL.KP])

If the absolute value of <error> is larger than PL.ERRWTHRESH for a consecutive period of 100 ms, then a warning is generated. If the absolute value of <error> 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.

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 (pg 810) | PL.ERRFTHRESH (pg 811) | PL.ERRWTHRESH (pg 815)
Position Loop (pg 139)

25.30.5 PL.ERRWTHRESH

General Information	
Type	NV Parameter
Description	Sets the position error warning level.
Units	Depends on or 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 (pg 761) mm 0.000 to 1,192,878,014.464*MOTOR.PITCH (pg 761) 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 (pg 810)
Start Version	M_01-00-00-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3483h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	584	Yes	64 bit	No	M_01-03-00-000

Description

If this value is not equal 0 and the position error PL.ERR (pg 810) 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 (pg 810) | PL.ERRFTHRESH (pg 811) | PL.ERRMODE (pg 813)

13.5 Position Loop

25.30.6 PL.FB

General Information	
Type	R/O Parameter
Description	Reads the position feedback value.
Units	Depends on or 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	Float
See Also	FB1.OFFSET
Start Version	M_01-00-00-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	6064h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	588	Yes	64 bit	Yes	M_01-03-00-000

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.

AKD BASIC Note

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 by the user. If a new value is written to MOVE.P-OSCOMMAND then PL.FB will be automatically changed such that PL.ERROR (the difference between them) is unchanged.

Related Topics

PL.ERR (pg 810) | PL.ERRFTHRESH (pg 811) | PL.ERRMODE (pg 813) | PL.ERRWTHRESH (pg 815)

13.5 Position Loop

25.30.7 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-x-xxxxx-NAxx-xxxx) 0 to 2 (for AKD-x-xxxxx-NBxx-xxxx)
Default Value	0
Data Type	Integer
See Also	VL.FBSOURCE
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	592	No	8 bit	No	M_01-03-00-000

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-x-xxxxx-NBxx-xxxx).

Related Topics

PL.ERR (pg 810) | PL.ERRFTHRESH (pg 811) | PL.ERRMODE (pg 813) | PL.ERRWTHRESH (pg 815)
Position Loop (pg 139)

25.30.8 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 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.

Related Topics

Electronic Gearing (pg 117)

25.30.9 PL.INTINMAX

General Information	
Type	NV Parameter
Description	Limits the input of the position loop integrator by setting the input saturation.
Units	Depends on or 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 (pg 761) mm 0.000 to 4,294,967.296*MOTOR.PITCH (pg 761) 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 (pg 761) mm 9MOTOR.PITCH (pg 761) 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3481h/1	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	594	Yes	64 bit	No	M_01-03-00-000

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 (pg 810) | PL.ERRFTHRESH (pg 811) | PL.ERRMODE (pg 813) | PL.ERRWTHRESH (pg 815)
13.5   Position Loop

25.30.10 PL.INTOUTMAX

General Information	
Type	NV Parameter
Description	Limits the output of the position loop integrator by setting the output saturation.
Units	Depends on or 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 counts16 bit Linear: 0.000 to 18,446,744,073,709.000 counts 0.000 to 4,294.968*MOTOR.PITCH (pg 761) 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 (pg 761) mm 9MOTOR.PITCH (pg 761) 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3481h/2	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	598	Yes	64 bit	No	M_01-03-00-000

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 (pg 810) | PL.ERRFTHRESH (pg 811) | PL.ERRMODE (pg 813) | PL.ERRWTHRESH (pg 815)

13.5 Position Loop

25.30.11 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3480h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	602	No	32 bit	No	M_01-03-00-000

Description

PL.KI sets the integral gain of the position regulator PID loop.

Related Topics

PL.ERR (pg 810) | PL.ERRFTHRESH (pg 811) | PL.ERRMODE (pg 813) | PL.ERRWTHRESH (pg 815)

13.5 Position Loop

25.30.12 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 (pg 824),
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3542h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	604	No	32 bit	No	M_01-03-00-000

Description

PL.KP sets the proportional gain of the position regulator PID loop.

Related Topics

PL.ERR (pg 810) | PL.ERRFTHRESH (pg 811) | PL.ERRMODE (pg 813) | PL.ERRWTHRESH (pg 815)

13.5 Position Loop

25.30.13 PL.MODP1

General Information	
Type	R/W parameter
Description	Sets modulo range parameter.
Units	Depends on UNIT.PROTARY (pg 908) and UNIT.PLINEAR (pg 906)
Range	N/A
Default Value	N/A
Data Type	Float
See Also	N/A
Start Version	M_01-00-00-000

AKD BASIC Information	
Type	R/O
Range	0

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open		M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	604	Yes	64 bit	Yes	M_01-03-00-000

Description

This parameter is either the beginning or the end of the modulo range, depending on whether this value is smaller or larger than PL.MODP2. If you set PL.MODP1 equal to PL.MODP2, an error message occurs.

Condition	Beginning of the modulo-range	End of the modulo-range
PL.MODP1 < PL.MODP2	PL.MODP1	PL.MODP2
PL.MODP2 < PL.MODP1	PL.MODP2	PL.MODP1

Related Topics

PL.ERR (pg 810) | PL.ERRFTHRESH (pg 811) | PL.ERRMODE (pg 813) | PL.ERRWTHRESH (pg 815)

Position Loop (pg 139)

25.30.14 PL.MODP2

General Information	
Type	R/W Parameter
Description	Sets the beginning or end modulo range parameter.
Units	Depends on and .
Range	N/A
Default Value	N/A
Data Type	Float
See Also	N/A
Start Version	M_01-00-00-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3638h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	610	Yes	64 bit	Yes	M_01-03-00-000

Description

This parameter is either the beginning or the end of the modulo range, depending on whether this value is smaller or larger than PL.MODP1 (pg 826).

Condition	Beginning of the modulo range	End of the modulo range
PL.MODP1 < PL.MODP2	PL.MODP1	PL.MODP2
PL.MODP2 < PL.MODP1	PL.MODP2	PL.MODP1

Related Topics

PL.ERR (pg 810) | PL.ERRFTHRESH (pg 811) | PL.ERRMODE (pg 813) | PL.ERRWTHRESH (pg 815)

13.5 Position Loop

25.30.15 PL.MODPDIR

General Information	
Type	R/W Parameter
Description	Sets the direction for absolute motion tasks.
Units	N/A
Range	0 to 2
Default Value	0
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3430h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	614	No	8 bit	No	M_01-03-00-000

Description

This parameter defines the direction of an absolute motion task when the modulo position has been activated. For more details about absolute motion tasks, see .. For absolute motion tasks, you can only select a target position within the modulo range.

PL.MODPDIR Settings

Value	Motion	Description
0	Inside Range	The motor moves in a negative direction if the target position of the absolute motion task is less than the current position. The motor moves in positive direction if the target position of the absolute motion task is greater than the current position.
1	Positive	The motor always moves in a positive direction relative to the target position of the absolute motion task.
2	Negative	The motor always moves in a negative direction relative to the target position of the absolute motion task.
3	Shortest Distance	The motor always moves the shortest distance in order to reach the target position within the modulo-range.

25.30.16 PL.MODPEN

General Information	
Type	R/W Parameter
Description	Enables the modulo position.
Units	N/A
Range	0 to 1
Default Value	0
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	35CFh/0	M_01-00-00-000

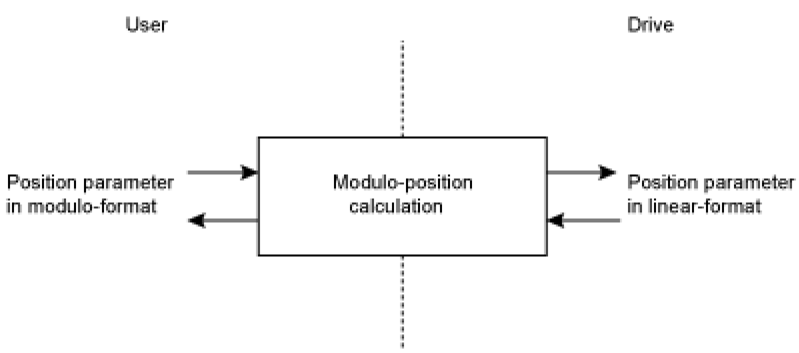
Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	616	No	8 bit	No	M_01-03-00-000

Description

A value of 0 disables the modulo-position and a value of 1 enables the modulo-position feature. The modulo-position feature can be used for circular applications such as round tables.

The position loop of the drive uses always a linear position variable but the data-exchange between the user and the drive uses the modulo-position calculation in order to convert values from linear format into modulo format and vice versa.

The following figure shows the interface between the user and the drive for PL.MODPEN=1:



Related Topics

Position Loop (pg 139)

25.31 PLS Parameters

This section describes the PLS parameters.

25.31.1 PLS.EN

General Information	
Type	R/W Parameter
Description	Enables programmable limit switch (PLS).
Units	N/A
Range	0 to 255
Default Value	0
Data Type	Integer
See Also	PLS.MODE (pg 832), PLS.RESET, PLS.STATE, PLS.UNITS, PLS.P1 to PLS.P8, PLS.WIDTH1 to PLS.WIDTH8, PLS.T1 to PLS.T8
Start Version	M_01-02-03-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	34A3h/1	M_01-02-03-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	618	No	16 bit	No	M_01-03-00-000

Description

PLS.EN is a bit variable which determines the mode of an individual PLS. Eight PLSs are available in the drive.

Example

Bit Value	Behavior
Bit 0 = 0	Disables PLS 1
Bit 0 = 1	Enables PLS 1
Bit 7 = 0	Disables PLS 8
Bit 7 = 1	Enables PLS 8

Related Topics

Programmable Limit Switch (pg 120)

25.31.2 PLS.MODE

General Information	
Type	R/W Parameter
Description	Selects programmable limit switch mode.
Units	N/A
Range	0 to 255
Default Value	0
Data Type	Integer
See Also	PLS.EN, PLS.RESET, PLS.STATE, PLS.UNITS, PLS.P1 to PLS.P8, PLS.WIDTH1 to PLS.WIDTH8, PLS.T1 to PLS.T8
Start Version	M_01-02-03-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	34A3h/3	M_01-02-03-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	620	No	16 bit	No	M_01-03-00-000

Description

PLS.MODE is a bit variable which determines the mode of an individual PLS. Eight PLSs are available in the drive.

Example

Bit Value	Behavior
Bit 0 = 0	PLS 1 is monitored continuously.
Bit 0 = 1	PLS 1 is monitored until it is triggered once (single-shot method). The PLS observation can be re-armed using the PLS.RESET command.
Bit 7 = 0	PLS 8 is monitored continuously.
Bit 7 = 1	PLS 8 is monitored until it is triggered once (single-shot method). The PLS observation can be re-armed using the PLS.RESET command.

Related Topics

Programmable Limit Switch (pg 120)

25.31.3 PLS.P1 TO PLS.P8

General Information	
Type	R/W Parameter
Description	Sets the trigger point for programmable limit switches.
Units	Depends on UNIT.PROTARY or UNIT.PLINEAR
Range	N/A
Default Value	0
Data Type	Float
See Also	UNIT.PROTARY (pg 908)
Start Version	M_01-02-03-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	34A0h/1	PLS.P1	M_01-02-03-000
	34A0h/2	PLS.P2	
	34A0h/3	PLS.P3	
	34A0h/4	PLS.P4	
	34A0h/5	PLS.P5	
	34A0h/6	PLS.P6	
	34A0h/7	PLS.P7	
	34A0h/8	PLS.P8	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	622	PLS.P1	Yes	64 bit	Yes	M_01-03-00-000
	626	PLS.P2				
	630	PLS.P3				
	634	PLS.P4				
	638	PLS.P5				
	642	PLS.P6				
	646	PLS.P7				
	650	PLS.P8				

Description

PLS.P1 to PLS.P8 define the trigger point of the PLS. For further information about how these parameters affect PLS behavior, see the PLS.UNITS parameter description.

Related Topics

Programmable Limit Switch (pg 120)

25.31.4 PLS.RESET

General Information	
Type	R/O Parameter
Description	Resets programmable limit switch.
Units	N/A
Range	0 to 255
Default Value	N/A
Data Type	Integer
See Also	PLS.EN, PLS.MODE, PLS.STATE, PLS.UNITS, PLS.Px (x=1…8), PLS.WIDTHx (x=1…8), PLS.Tx (x=1…8)
Start Version	M_01-02-03-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	34A3h/2	M_01-02-03-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	654	No	16 bit	No	M_01-03-00-000

Description

This parameter is a bit variable and is used in order to re-arm the corresponding PLS.STATE observation for another single-shot PLS use (see also PLS.MODE).

Example

Bit Value	Behavior
Bit 0 = 0	The PLS 1 observation (PLS.STATE bit 0) is not re-armed.
Bit 0 = 1	The PLS 1 observation (PLS.STATE bit 0) is re-armed.
Bit 7 = 0	The PLS 8 observation (PLS.STATE bit 7) is not re-armed.
Bit 7 = 1	The PLS 8 observation (PLS.STATE bit 7) is re-armed.

Related Topics

Programmable Limit Switch (pg 120)

25.31.5 PLS.STATE

General Information	
Type	R/O Parameter
Description	Reads the programmable limit switch state.
Units	N/A
Range	N/A
Default Value	N/A
Data Type	Integer
See Also	PLS.EN, PLS.RESET, PLS.UNITS, PLS.MODE, PLS.P1 TO PLS.P8, PLS.WIDTH1 TO PLS.WIDTH8, PLS.T1 TO PLS.T8
Start Version	M_01-02-03-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	34A3h/4	M_01-02-03-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	656	No	16 bit	No	M_01-03-00-000

Description

This parameter is a bit variable and displays the current status of the individual programmable limit switches.

Example

Bit 0 = 0: Programmable Limit Switch 1 (PLS 1) is not active.

Bit 0 = 1: Programmable Limit Switch 1 (PLS 1) is active.

Bit 7 = 0: Programmable Limit Switch 8 (PLS 8) is not active.

Bit 7 = 1: Programmable Limit Switch 8 (PLS 8) is not active.

Related Topics

Programmable Limit Switch (pg 120)

25.31.6 PLS.T1 TO PLS.T8

General Information	
Type	R/W parameter
Description	Sets programmable limit switch time
Units	ms
Range	0 to 65,536 ms
Default Value	500 ms
Data Type	Integer
See Also	PLS.EN, PLS.RESET, PLS.STATE, PLS.UNITS, PLS.MODE, PLS.WIDTH1 TO PLS.WIDTH8, PLS.P1 TO PLS.P8
Start Version	M_01-02-03-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	34A2h/1	PLS.T1	M_01-02-03-000
	34A2h/2	PLS.T2	
	34A2h/3	PLS.T3	
	34A2h/4	PLS.T4	
	34A2h/5	PLS.T5	
	34A2h/6	PLS.T6	
	34A2h/7	PLS.T7	
	34A2h/8	PLS.T8	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	658	PLS.T1	No	16 bit	No	M_01-03-00-000
	660	PLS.T2				
	662	PLS.T3				
	664	PLS.T4				
	666	PLS.T5				
	668	PLS.T6				
	670	PLS.T7				
	672	PLS.T8				

Description

These parameters define the time of the PLS pulse for time-based PLS handling.

For further information about the PLS functionality, especially the meaning of the PLS.T1 to PLS.T8 parameter, refer to the PLS.UNITS parameter.

Related Topics

Programmable Limit Switch (pg 120)

25.31.7 PLS.UNITS

General Information	
Type	R/W parameter
Description	Sets programmable limit switch (PLS) units.
Units	N/A
Range	0 to 255
Default Value	0
Data Type	Integer
See Also	PLS.EN (pg 831), PLS.RESET (pg 835), PLS.STATE (pg 836), PLS.MODE (pg 832), PLS.P1 TO PLS.P8 (pg 833)PLS.WIDTH1 TO PLS.WIDTH8 (pg 841), PLS.T1 TO PLS.T8 (pg 837)
Start Version	M_01-02-03-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	34A4h/0	M_01-02-03-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	674	No	8 bit	No	M_01-03-00-000

Description

PLS.UNITS is a bit variable which determines the behavior of the eight PLSs available in the drive. This parameter is used to select the units for the PLS pulse.

Examples

Bit Value	Behavior
Bit 0 = 0 Position-based PLS handling.	The PLS.STATE parameter displays an active PLS 1 when the position is within the range of PLS.P1 + PLS.WIDTH1 (PLS.P1 <= PL.FB <= PLS.P1+PLS.WIDTH1). When the parameter PLS.WIDTH1 has been set to the value of 0, this bit will be activated as soon as PLS.FB >= PL.P1.
Bit 0 = 1 Time-based PLS handling.	After PLS.P1 is crossed, the PLS.STATE parameter displays an active PLS 1 for a PLS.T1 ms period of time.

Bit Value	Behavior
Bit 7 = 0 Position-based PLS handling.	The PLS.STATE parameter displays an active PLS 8 when the position is within the range of PLS.P8 + PLS.WIDTH8 (PLS.P8 <= PL.FB <= PLS.P8+PLS.WIDTH8). When the parameter PLS.WIDTH8 has been set to the value of 0, this bit will be activated as soon as PLS.FB >= PL.P8.
Bit 7 = 1 Time-based PLS handling.	After PLS.P8 has been crossed. the PLS.STATE parameter displays an active PLS 8 for a PLS.T8 ms period of time.

Continuous position-based PLS handling

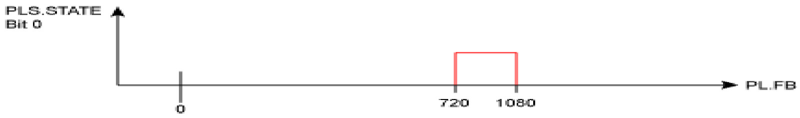
PLS.P1 = 720

PLS.WIDTH1 = 360

PLS.UNITS bit 0 (for PLS 1) = low; PLS.T1 is not considered.

PLS.EN bit 0 (for PLS 1) = high

PLS.MODE bit 0 (for PLS 1) = low



Time-based PLS handling

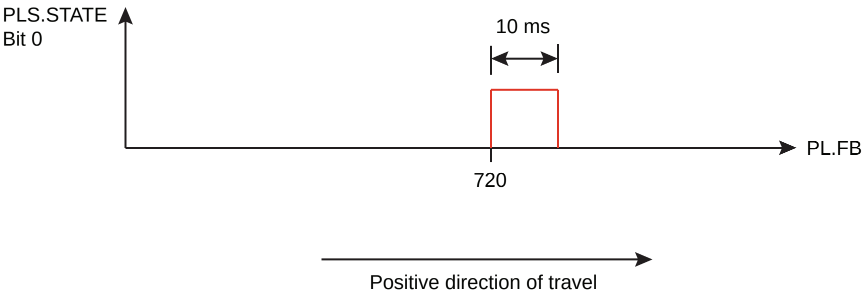
PLS.P1 = 720

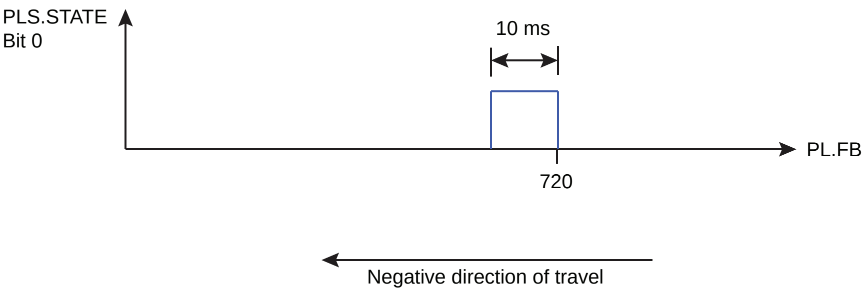
PLS.T1 = 10

PLS.UNITS bit 0 (for PLS 1) = low; PLS.WIDTH1 is not considered.

PLS.EN bit 0 (for PLS 1) = high

PLS.MODE bit 0 (for PLS 1) = low





Related Topics

Programmable Limit Switch (pg 120)

25.31.8 PLS.WIDTH1 TO PLS.WIDTH8

General Information	
Type	R/W parameter
Description	Programmable Limit Switch Width
Units	Depends on UNIT.PROTARY or UNIT.PLINEAR
Range	N/A
Default Value	0
Data Type	Float
See Also	PLS.EN, PLS.RESET, PLS.STATE, PLS.UNITS, PLS.MODE, PLS.P1 TO PLS.P8, PLS.T1 TO PLS T8
Start Version	M_01-02-03-000

AKD BASIC Information	
Data Type	Integer

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	34A1h/1	PLS.WIDTH1	M_01-02-03-000
	34A1h/2	PLS.WIDTH2	
	34A1h/3	PLS.WIDTH3	
	34A1h/4	PLS.WIDTH4	
	34A1h/5	PLS.WIDTH5	
	34A1h/6	PLS.WIDTH6	
	34A1h/7	PLS.WIDTH7	
	34A1h/8	PLS.WIDTH8	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	676	PLS.WIDTH1	Yes	64 bit	Yes	M_01-03-00-000
	680	PLS.WIDTH2				
	684	PLS.WIDTH3				
	688	PLS.WIDTH4				
	692	PLS.WIDTH5				
	696	PLS.WIDTH6				
	700	PLS.WIDTH7				
	704	PLS.WIDTH8				

Description

These parameter define the width of the PLS pulse for position-based PLS handling. For further information about the PLS functionality, especially the meaning of the PLS.WIDTH1 to PLS.WIDTH8 parameter, refer to the PLS.UNITS parameter.

Related Topics

Programmable Limit Switch (pg 120)

25.32 REC Parameters

This section describes the REC parameters.

25.32.1 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 (pg 846), REC.OFF (pg 849)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	708	No	8 bit	No	M_01-03-00-000

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 (pg 228)

25.32.2 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 (pg 858)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	N/A
AKD EtherNet/IP	N/A

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 (pg 850).
- Set an internal value or variable of the drive (same as for DRV.MEMADDR (pg 539) input).

Related Topics

Scope (pg 228)

25.32.3 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	710	No	8 bit	No	M_01-03-00-000

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 (pg 228)

25.32.4 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 (pg 858)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	712	No	16 bit	No	M_01-03-00-000

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 µs.

Related Topics

Scope (pg 228)

25.32.5 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 (pg 858)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	714	No	16 bit	No	M_01-03-00-000

Description

REC.NUMPOINTS specifies the number of points (samples) to record.

Related Topics

Scope (pg 228)

25.32.6 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	716	No	Command	No	M_01-03-00-000

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 (pg 228)

REC.READY

25.32.7 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 (pg 845)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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 (pg 228)

25.32.8 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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<=6):

```
Recording
<N>,<G>
<parameter name 1> … <parameter name M>
Value11 … Value1M
Value N1 …  ValueNM
```

Related Topics

Scope (pg 228)

25.32.9 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.R-ETRIEVESIZE
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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.

WorkBench 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 (pg 858) 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

Scope (pg 228)

25.32.10 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 (pg 851), REC.RETRIEVEDATA (pg 852)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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). WorkBench 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 (pg 228)

25.32.11 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

This command retrieves the recorded header without the data of the recording.

WorkBench 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 (pg 228)

25.32.12 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 (pg 852), REC.RETRIEVEHDR (pg 855)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

This parameter sets the number of samples that REC.RETRIEVEDATA (pg 852) returns.

WorkBench also uses this parameter to set the number of samples returned when retrieving the data continuously for RT recoding.

Use REC.RETRIEVE (pg 851) for the complete recording information view.

Related Topics

Scope (pg 228)

25.32.13 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.R-ETRIEVESIZE
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	718	No	8 bit	No	M_01-03-00-000

Description

This parameter sets the stop type for the recording.

0 = Recorder runs, continuously filling the recording circular buffer.

1 = Recorder fills in the buffer once.

To stop RT recording, execute REC.OFF.

Related Topics

Scope (pg 228)

25.32.14 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	720	No	Command	No	M_01-03-00-000

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 (pg 228)

25.32.15 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

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 (pg 810), 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 (pg 228)

25.32.16 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 (pg 858), REC.NUMPOINTS (pg 848)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	722	No	8 bit	No	M_01-03-00-000

Description

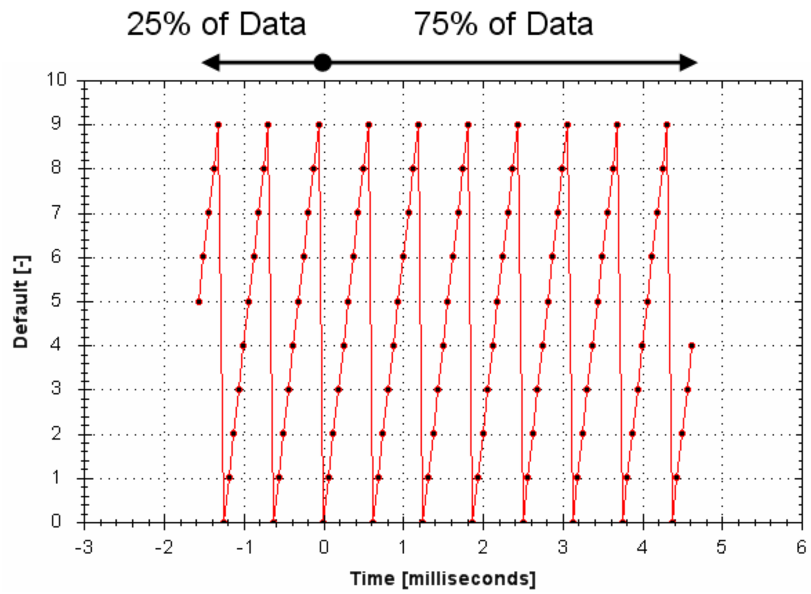
REC.TRIGPOS sets the trigger position in the recording buffer. The recording buffer size is defined by REC.NUMPOINTS. The input value is a percentage of the buffer (that is, a value of 25 means saving 25% of the buffer data before the trigger occurs and 75% after it occurs). This parameter is only used 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 WorkBench 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.



Related Topics

Scope (pg 228)

25.32.17 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

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 (pg 228)

25.32.18 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	726	No	8 bit	No	M_01-03-00-000

Description

REC.TRIGSLOPE sets the recorder trigger slope. This parameter is only used when REC.TRIGTYPE = 2 or 3.

Related Topics

Scope (pg 228)

25.32.19 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 (pg 858), REC.TRIGPARAM, REC.TRIGVAL, REC.TRIGSLOPE REC.TRIGPOS
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	728	No	8 bit	No	M_01-03-00-000

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 (pg 228)

25.32.20 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 (pg 858), REC.TRIGPARAM, REC.TRIGVAL, REC.TRIGSLOPE, REC.TRIGPOS
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	730	Yes	64 bit	Yes	M_01-03-00-000

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 (pg 228)

25.33 REGEN Parameters

This section describes the REGEN parameters.

25.33.1 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3416h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	734	Yes	64 bit	No	M_01-03-00-000

Description

This parameter reads regen resistor's calculated power, which is determined as follows:

$(V^2 / R) * \mathrm{DutyCycle}$

Related Topics

7.2 Regeneration

25.33.2 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 Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1736	Yes	64-bit	No

Description

This parameter returns a filtered value of REGEN.POWER. A single order lowpass of 1/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 Frequence = 1/10 seconds = 0.1 Hz

REGEN.POWERFILTERED = REGEN.POWER * 0.1Hz lowpass

Related Topics

REGEN.POWER (pg 867) | REGEN.TEXT (pg 870) | REGEN.WATTEXT (pg 873)

25.33.3 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	35C2h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	738	No	16 bit	No	M_01-03-00-000

Description

REGEN.REXT sets the external user-defined regen resistor resistance. This variable is needed for the regen resistor temperature estimation algorithm.

Related Topics

7.2 Regeneration

25.33.4 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 (pg 873), REGEN.REXT (pg 869)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3415h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	740	No	32 bit	No	M_01-03-00-000

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:

REGEN.TEXT = (1.1)*((E in joules)/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

7.2 Regeneration

25.33.5 REGEN.TYPE

General Information	
Type	NV Parameter
Function	Sets the regen resistor type.
WorkBench Location (Screen/Dialog Box)	Power/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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3412h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	742	No	8 bit	Yes	M_01-03-00-000

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

7.2 Regeneration

25.33.6 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3414h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	744	No	16 bit	No	M_01-03-00-000

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 wil be 0 and a fault will be issued.

Related Topics

7.2 Regeneration

25.34 SD Commands

This section describes commands for SD Card functions.

25.34.1 SD.LOAD

General Information	
Type	Command
Description	Loads the drive state (BASIC program and NV parameters)from the SD card to the AKD (AKDs equipped with IO option card only).
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
Start Version	M_01-06-03-000

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1684	No	Command	No	M_01-06-03-000

Description

SD.LOAD will load the drive state from the SD card (if one exists) to the AKD. The files loaded to the drive include the BASIC binary file (program.bin) and the NV parameter file (drive.akd). These files must have these exact names or the drive will not recognize them.

Note that this command can only be executed when the drive is in the idle state (i.e. a program is not running) and the drive is disabled.

If a computer is not connected to the drive, the SD.LOAD command can also be issued using the rotary switches S1 and S2.

To load the SD drive state onto the AKD:

1. Set S1 to position 1
2. Set S2 to position 0
3. Hold down the B1 button on the top of the drive for 5 seconds.

NOTE While the load operation is completing the LED display will flash **Sd**. If an error occurs, the letter E followed by three numbers will flash on the display. See for a description of SD Errors.

Related Topics

SD.SAVE (pg 876)

25.34.2 SD.SAVE

General Information	
Type	Command
Description	Saves the drive state (BASIC program and NV parameters) to the SD card (AKDs equipped with IO option card only).
Units	N/A
Range	N/A
Default Value	N/A
Data Type	N/A
Start Version	M_01-06-03-000

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1686	No	Command	No	M_01-06-03-000

Description

SD.SAVE will copy the state of a BASIC drive to the SD card (if one exists). The state of the drive consists of the BASIC binary file and NV parameters. Before an SD.Save is performed, the drive will perform a DRV.NVLOAD and return all of the drive parameters to their NV state. The DRV.NVLOAD is necessary in order to capture the NV parameter states for the parameter file(drive.akd). The files saved to the SD card include the BASIC binary file (program.bin) and the NV parameter file (drive.akd). In order for a drive to recognize and load these files, they must be named program.bin and drive.akd.

NOTE This command can only be executed when the drive is in the idle state (i.e. a program is not running) and the drive is disabled.

If a computer is not connected to the drive, the SD.SAVE command can also be issued using the rotary switches S1 and S2.

To save the drive state onto the SD card using rotary switches:

1. Set S1 to position 1
2. Set S2 to position 1
3. Hold down the B1 button on the top of the drive for 5 seconds.

NOTE While the load operation is completing the LED display will flash **Sd**. If an error occurs, the letter E followed by three numbers will flash on the display. See for a description of SD Errors.

Related Topics

SD.LOAD (pg 875)

25.34.3 SD.STATUS

General Information	
Type	R/O
Description	Reads the status of the SD card.
Units	N/A
Range	0 to 2
Default Value	0
Data Type	Integer
Start Version	M_01-06-03-000

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1688	No	8-bit	No	M_01-06-03-000

Description

This parameter reads the status of the SD card.

Status	Description
0	Ready – an SD card is inserted into the drive and you can read and write to it.
1	Read Only – an SD card is inserted into the drive and the write protection tab on the card prohibits writes.
2	Not Inserted – an SD card is not inserted into the drive.

Related Topics

SD.LOAD (pg 875) | SD.SAVE (pg 876)

25.35 SM Parameters

This section describes the SM parameters.

25.35.1 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	746	No	32 bit	Yes	M_01-03-00-000

Description

SM.I1 defines the current that is used in service motion modes 0 and 1 (see SM.MODE (pg 881)).

Related Topics

Service Motion (pg 164)

25.35.2 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	748	No	32 bit	Yes	M_01-03-00-000

Description

SM.I2 defines the current that is used in service motion mode 1 (see SM.MODE (pg 881)).

Related Topics

Service Motion (pg 164)

25.35.3 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 (pg 879), SM.I2 (pg 880), SM.MOVE (pg 884), SM.T1 (pg 885) SM.T2 (pg 886), SM.V1 (pg 887), SM.V2 (pg 889), DRV.ACC (pg 489)DRV.DEC (pg 501)
Start Version	M_01-01-00-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	750	No	16 bit	No	M_01-03-00-000

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:

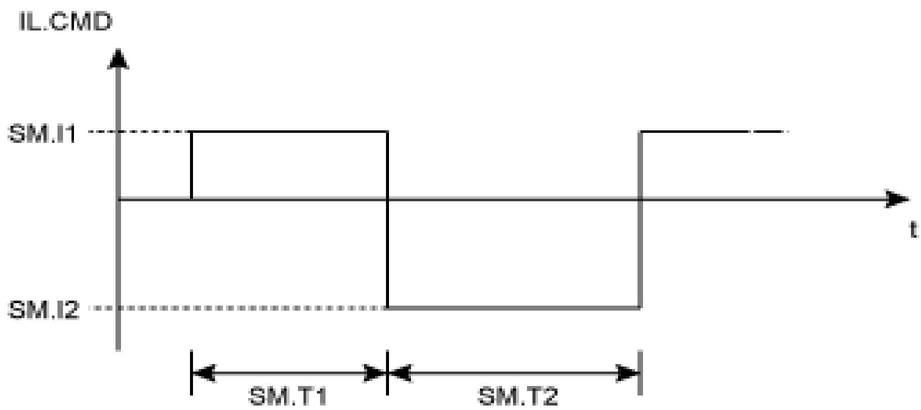
SM.MODE	Description	Requirements
0	Constant motion in closed current loop mode of operation. • DRV.OPMODE 0: The drive generates a constant current command value (SM.I1) for a certain amount of time (if SM.T1>0) or endless (if SM.T1=0). The drive will not generate any ramps in this mode of operation. • 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. The service motion can be stopped by using the DRV.STOP command.	DRV.OPMODE = 0,1, or 2 DRV.CMDSOURCE = 0

SM.MODE	Description	Requirements
1	• 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

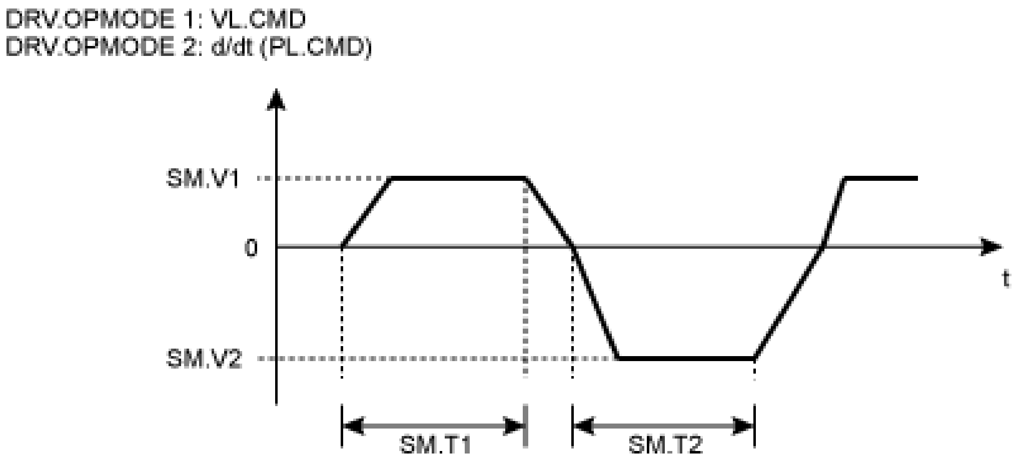
Ramps

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



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 (pg 164)

25.35.4 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	752	No	Command	No	M_01-03-00-000

Description

This command starts the service motion that has been selected by the SM.MODE parameter.

Related Topics

Service Motion (pg 164)

25.35.5 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	754	No	16 bit	No	M_01-03-00-000

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 (pg 164)

25.35.6 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	756	No	16 bit	No	M_01-03-00-000

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 (pg 164)

25.35.7 SM.V1

General Information	
Type	NV Parameter
Description	Sets service motion velocity 1; active in opmode 1 (velocity) and 2 (position).
Units	Depends on or Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: Counts/s, mm/s, µm/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 (pg 761) to 250.000*MOTOR.PITCH (pg 761) mm/s -250,000.000*MOTOR.PITCH (pg 761) to 250,000.000*MOTOR.PITCH (pg 761) µm/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 (pg 761) mm/s 999.998*MOTOR.PITCH (pg 761) µm/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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	758	Yes	Low 32 bit word	Yes	M_01-03-00-000

Description

SM.V1 defines the velocity that is used in service motion modes 0 and 1 (see SM.MODE (pg 881)) in the closed velocity and position mode of operation.

Related Topics

14.3 Service Motion

25.35.8 SM.V2

General Information	
Type	NV Parameter
Description	Sets service motion velocity 2; active in opmode 1 (velocity) and 2 (position).
Units	Depends on or Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: counts/s, mm/s, µm/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 (pg 761) to 250.000*MOTOR.PITCH (pg 761) mm/s -250,000.000*MOTOR.PITCH (pg 761) to 250,000.000*MOTOR.PITCH (pg 761) µm/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 (pg 761) mm/s -999.998*MOTOR.PITCH (pg 761) µm/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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	760	Yes	Low 32 bit word	Yes	M_01-03-00-000

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

14.3 Service Motion

25.36 STO Parameters

This section describes the STO parameters.

25.36.1 STO.STATE

General Information	
Type	R/O Parameter
Description	Returns the status of the safe torque off.
Units	N/A
Range	0 to 1
Default Value	N/A
Data Type	Integer
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	762	No	8 bit	No	M_01-03-00-000

Description

STO.STATE returns the status of the safe torque off.

1 - Safe torque on (no safe torque off fault).

0 - Safe torque off (safe torque off fault).

Related Topics

12.6 Limits

25.37 SWLS Parameters

This section describes the SWLS parameters.

25.37.1 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	25.13.46 DRV.MOTIONSTAT
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	764	No	16 bit	No	M_01-03-00-000

Description

This parameter enables the software travel limit switches. The software limit switches are only active if the axis is homed.

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

12.6 Limits

25.37.2 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
See Also	,
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	607Dh/1	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	766	Yes	64 bit	Yes	M_01-03-00-000

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. Whichever switch is set largest is the positive limit switch; the other switch becomes the negative 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

12.6 Limits

25.37.3 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
See Also	,
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	607Dh/2	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	770	Yes	64 bit	Yes	M_01-03-00-000

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. Whichever switch is set largest is the positive limit switch; the other switch becomes the negative 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

12.6 Limits

25.13.46 DRV.MOTIONSTAT

25.37.4 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
See Also	N/A
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	774	No	16 bit	No	M_01-03-00-000

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 (pg 895) is not active.

Bit 0 = 1: SWLS.LIMIT0 is active.

Bit 1 = 0: SWLS.LIMIT1 (pg 896) is not active.

Bit 1 = 1: SWLS.LIMIT1 is active.

Bits 2 to 7 are currently not in use.

Related Topics

12.6 Limits

25.38 TEMP Parameters

This section describes the TEMP parameters.

25.38.1 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

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1748	No	16-bit	Yes

Description

Reads the temperature of the control board temperature sensor.

Related Topics

DRV.TEMPERATURES (pg 557) | TEMP.POWER1 to TEMP.POWER3 (pg 900)

25.38.2 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

Fieldbus Information

Fieldbus	Parameter	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	TEMP.POWER1	1750	No	16-bit	Yes
	TEMP.POWER2	1752			
	TEMP.POWER3	1753			

Description

Reads the temperature of the power board(s) temperature sensor(s).

Related Topics

DRV.TEMPERATURES (pg 557) | TEMP.CONTROL (pg 899)

25.39 UNIT Parameters

This section describes the UNIT parameters.

25.39.1 UNIT.ACCLINEAR

General Information	
Type	NV Parameter
Description	Sets the linear acceleration/deceleration units.
Units	N/A
Range	0 to 3
Default Value	0
Data Type	Integer
See Also	DRV.ACC (pg 489), DRV.DEC (pg 501), MOTOR.TYPE (pg 771)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	776	No	8 bit	No	M_01-03-00-000

Description

UNIT.ACCLINEAR sets the units type for the acceleration and deceleration parameters, when the motor type (MOTOR.TYPE (pg 771)) is linear.

Type	Description
0	[Custom Units (pg 96)]/s^2
1	millimeters per second squared (mm/s^2)
2	micrometers per second squared (µm/s^2)
3	Feedback counts/s^2

Related Topics

Selecting Units for Your Application (pg 95)

25.39.2 UNIT.ACCROTARY

General Information	
Type	NV Parameter
Description	Sets the rotary acceleration/deceleration units.
Units	rpm/s, rps/s, deg/s^2 , [Custom Units (pg 96)] /s^2
Range	0 to 3 rpm/s
Default Value	0 rpm/s
Data Type	Integer
See Also	DRV.ACC (pg 489), MOTOR.TYPE (pg 771)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3659h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	778	No	8 bit	No	M_01-03-00-000

Description

UNIT.ACCROTARY sets the acceleration/deceleration units when the motor type (MOTOR.TYPE (pg 771)) is rotary.

Type	Description
0	rpm/s
1	rps/s
2	deg/s^2
3	(Custom Units (pg 96)) /s^2

Related Topics

Selecting Units for Your Application (pg 95)

"DRV.DEC " (➔ p. 501)

25.39.3 UNIT.LABEL

General Information	
Type	NV Parameter
Description	Sets user-defined name for user-defined position units.
Units	N/A
Range	Maximum 16 characters, no spaces
Default Value	Custom Units (pg 96)
Data Type	String
See Also	UNIT.PLINEAR (pg 906), UNIT.POUT (pg 907)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

If you define a special position unit with UNIT.PLINEAR (pg 906) and UNIT.POUT (pg 907), then you can give this unit a descriptive name. You can name the unit anything you wish, as long as the name is limited to 16 characters and includes no spaces. The label used for velocity and acceleration are in terms of this descriptive name.

This parameter is descriptive only and does not influence drive internal functions in any way.

Related Topics

Selecting Units for Your Application (pg 95)

25.39.4 UNIT.PIN

General Information	
Type	NV Parameter
Description	Sets gear IN for the unit conversion.
Units	User units
Range	0 to 4,294,967,295
Default Value	100
Data Type	Integer
See Also	UNIT.POUT (pg 907)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	35CAh/0 6092h/1	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	780	No	32 bit	No	M_01-03-00-000

Description

UNIT.PIN is used in conjunction with UNIT.POUT (pg 907) to set application specific units. This parameter is used as follows in the drive unit conversion:

- For position, this parameter sets the units as [Custom Units (pg 96)]/rev.
- For velocity, this parameter sets the units as [Custom Units (pg 96)]/s.
- For acceleration/deceleration, this parameter sets the units as [Custom Units (pg 96)]/s^2.

Related Topics

Selecting Units for Your Application (pg 95)

25.39.5 UNIT.PLINEAR

General Information	
Type	NV Parameter
Description	Sets the linear position units.
Units	N/A
Range	0 to 4
Default Value	0
Data Type	Integer
See Also	PL.FB (pg 817), PL.CMD (pg 809), MOTOR.TYPE (pg 771)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	782	No	8 bit	No	M_01-03-00-000

Description

UNIT.PLINEAR sets the units type for the position parameters when the motor type (MOTOR.TYPE) is linear.

Type	Description
0	32-bit counts
1	Millimeters (mm)
2	Micrometers (µm)
3	(PIN/POUT) per revolution
4	16-bit counts

Related Topics

Selecting Units for Your Application (pg 95)

25.39.6 UNIT.POUT

General Information	
Type	NV Parameter
Description	Sets gear out for the unit conversion.
Units	User units.
Range	0 to 4,294,967,295
Default Value	20
Data Type	Integer
See Also	UNIT.PLINEAR (pg 906)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	35CBh/0 6092h/2	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	784	No	32 bit	No	M_01-03-00-000

Description

UNIT.POUT is used with UNIT.PIN (pg 905) to set application specific units in UNIT.POUT. This parameter is used as follows in the drive unit conversion:

- For position, this parameter sets the units as [Custom Units (pg 96)] /rev.
- For velocity, this parameter sets the units as [Custom Units (pg 96)]/s.
- For acceleration/deceleration, this parameter sets the units as [Custom Units (pg 96)]/s^2.

Related Topics

Selecting Units for Your Application (pg 95)

25.39.7 UNIT.PROTARY

General Information	
Type	NV Parameter
Description	Sets the position units when the motor type (MOTOR.TYPE (pg 771)) is rotary.
Units	counts, rad, deg, Custom Units (pg 96), 16-bit counts
Range	0 to 4
Default Value	4 16-bit counts (for firmware versions M_01-02-00-000 and above) 0 counts (for firmware version M_01-01-00-000)
Data Type	Integer
See Also	PL.FB (pg 817), PL.CMD (pg 809), MOTOR.TYPE (pg 771)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3660h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	786	No	8 bit	No	M_01-03-00-000

Description

UNIT.PROTARY sets the position units when the motor type (MOTOR.TYPE (pg 771)) is rotary.

Value	Units
0	counts
1	radians
2	degrees
3	Custom Units (pg 96)
4	16-bit counts

Related Topics

Selecting Units for Your Application (pg 95)

25.39.8 UNIT.VLINEAR

General Information	
Type	NV Parameter
Description	Sets the linear velocity units.
Units	N/A
Range	0 to 3
Default Value	0
Data Type	Integer
See Also	VL.FB (pg 934), VL.CMDU (pg 932), VL.CMD (pg 931), MOTOR.TYPE (pg 771)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	788	No	8 bit	No	M_01-03-00-000

Description

UNIT.VLINEAR sets the units type for the velocity parameters when the motor type (MOTOR.TYPE (pg 771)) is linear.

Type	Description
0	(Custom Units (pg 96)) per second
1	Micrometers per second
2	Millimeters per second
3	Counts per second

Related Topics

Selecting Units for Your Application (pg 95)

25.39.9 UNIT.VROTARY

General Information	
Type	NV Parameter
Description	Sets the velocity units when the motor type (MOTOR.TYPE (pg 771)) is rotary.
Units	rpm, rps, deg/s, (Custom Units (pg 96))/s
Range	0 to 3
Default Value	0 rpm
Data Type	Integer
See Also	VL.FB (pg 934), VL.CMDU (pg 932), VL.CMD (pg 931), MOTOR.TYPE (pg 771)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	365Fh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	790	No	8 bit	No	M_01-03-00-000

Description

UNIT.VROTARY sets the velocity units when the motor type (MOTOR.TYPE) is rotary.

Value	Units
0	rpm
1	rps
2	deg/s
3	(Custom Units (pg 96)) /s

Related Topics

Selecting Units for Your Application (pg 95)

25.40 VBUS Parameters

This section describes the VBUS parameters.

25.40.1 VBUS.HALFVOLT

General Information	
Type	NV Parameter
Description	Changing voltage thresholds for HV and MV 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1218	No	8 bit	No

Description

This parameter is used in order to change some specific voltage thresholds in order to allow HV (high voltage) Drives working with MV (medium voltage) thresholds and to allow MV (medium voltage) Drives working with LV (low voltage) thresholds.

This parameter has an impact on the following voltage-thresholds:

1) DC-bus over-voltage threshold (see VBUS.OVFTHRESH).

2) The regen-resistor enable/disable voltage thresholds.

3) The inrush-relay enable/disable voltage thresholds.

A power-cycle is needed after changing the value of VBUS.HALFVOLT and after 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.

The VBUS.HALFVOLT command takes only effect for a HV or MV Drive.

VBUS.HALFVOLT = 0: The original voltage thresholds are used for the functions mentioned above.

VBUS.HALFVOLT = 1: Setting the parameter to 1 on a HV Drive causes the AKD to use the voltage thresholds of a MV Drive for the functions mentioned above. Setting the parameter to 1 on a MV Drive causes the AKD to use the voltage thresholds of a LV Drive for the functions mentioned above.

The sequence must be as follows:

1) Change the value of VBUS.HALFVOLT.

2) Trigger a DRV.NVSAVE command.

3) Power cycle the Drive in order to activate the new configuration.

Note

The DC-bus under voltage fault threshold (see VBUS.UVFTHRESH) 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 is supplied with a lower DC-bus voltage than the rated voltage.

25.40.2 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	794	No	16 bit	No	M_01-03-00-000

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 (pg 59)

25.40.3 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	796	No	16 bit	No	M_01-03-00-000

Description

If VBUS.VALUE value exceeds VBUS.OVWTHRESH, then a warning is generated.

Related Topics

Regeneration (pg 59)

25.40.4 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	798	No	8 bit	No	M_01-03-00-000

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 (pg 59)

25.40.5 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	800	No	16 bit	No	M_01-03-00-000

Description

VBUS.UVFTHRESH sets the undervoltage fault level of the DC bus.

The default value is read from the EEPROM, but can by modified by the user and stored on the NV RAM. This value varies according to drive type.

Related Topics

Regeneration (pg 59)

25.40.6 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	802	No	8 bit	No	M_01-03-00-000

Description

This parameter indicates undervoltage (UV) mode.

When VBUS.UVMODE - 0, an undervoltage fault is issued whenever the DC bus goes below the under-voltage threshold.

When VBUS.UVMODE = 1, an undervoltage fault is issued whenever the DC bus goes below the under voltage threshold and the controller attempts to enable the drive (software or hardware enable).

Related Topics

Regeneration (pg 59)

25.40.7 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 (pg 916)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	804	No	16 bit	No	M_01-03-00-000

Description

If VBUS.VALUE value drops below VBUS.UVWTHRESH, then a warning is generated.

Related Topics

Regeneration (pg 59)

25.40.8 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	361Ah/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	806	No	32 bit	No	M_01-03-00-000

Description

VBUS.VALUE reads the DC bus voltage.

Related Topics

7.2 Regeneration

25.41 VL Parameters

This section describes the VL parameters.

25.41.1 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 (pg 923), VL.ARZF1 TO VL.ARZF4 (pg 926), Sets the Q of the zero (numerator) of anti-resonance filter #1; active in opmodes 1 (velocity) and 2 (position) only. (pg 928)
Start Version	M_01-02-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	3406h/1	VL.ARPF1	M_01-02-00-000
	3406h/2	VL.ARPF2	
	3406h/3	VL.ARPF3	
	3406h/4	VL.ARPF4	

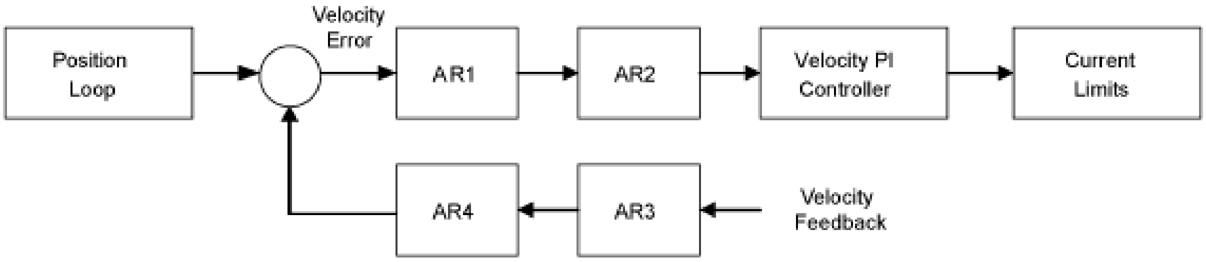
Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	808	VL.ARPF1	No	32 bit	No	M_01-03-00-000
	810	VL.ARPF2				
	812	VL.ARPF3				
	814	VL.ARPF4				

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:

$$ARx(\mathbf{s}) = [\mathbf{s}^2/(2\pi F_Z)^2 + \mathbf{s}/(Q_Z 2\pi F_Z) + 1]/ [\mathbf{s}^2/(2\pi F_P)^2 + \mathbf{s}/(Q_P 2\pi F_P) + 1]$$

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:

$$\mathbf{s} \approx (1-z^{-1})/t, \text{ where } t = 62.5\ \mu s$$

The poles are prewarped to F_P and the zeros are prewarped to F_Z.

Related Topics

Velocity Loop (pg 137)

25.41.2 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 (pg 921), VL.ARZF1 TO VL.ARZF4 (pg 926), VL.ARZQ1 TO VL.ARZQ4 (pg 928)
Start Version	M_01-02-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	3406h/5	VL.ARPQ1	M_01-02-00-000
	3406h/6	VL.ARPQ2	
	3406h/7	VL.ARPQ3	
	3406h/8	VL.ARPQ4	

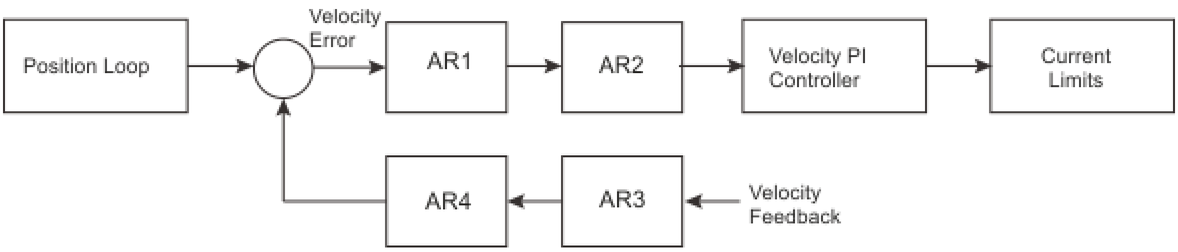
Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	816	VL.ARPQ1	No	32 bit	No	M_01-03-00-000
	818	VL.ARPQ2				
	820	VL.ARPQ3				
	822	VL.ARPQ4				

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:

$$ARx(\mathbf{s}) = [\mathbf{s}^2/(2\pi F_Z)^2 + \mathbf{s}/(Q_Z 2\pi F_Z) + 1] / [\mathbf{s}^2/(2\pi F_P)^2 + \mathbf{s}/(Q_P 2\pi F_P) + 1]$$

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:

$$\mathbf{s} \approx (1-z^{-1})/t, \text{ where } t = 62.5\ \mu s$$

The poles are prewarped to F_P and the zeros are prewarped to F_Z.

Related Topics

Velocity Loop (pg 137)

25.41.3 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	3405h/1	VL.ARTYPE1	M_01-02-00-000
	3405h/2	VL.ARTYPE2	
	3405h/3	VL.ARTYPE3	
	3405h/4	VL.ARTYPE4	

Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	824	VL.ARTYPE1	No	8 bit	No	M_01-03-00-000
	826	VL.ARTYPE2				
	828	VL.ARTYPE3				
	830	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

13.4 Velocity Loop

25.41.4 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 (pg 921),VL.ARPQ1 TO VL.ARPQ4 (pg 923), VL.ARZQ1 TO VL.ARZQ4 (pg 928)
Start Version	M_01-02-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex		Object Start Version
EtherCAT COE and CANopen	3406h/9	VL.ARZF1	M_01-02-00-000
	3406h/A	VL.ARZF2	
	3406h/B	VL.ARZF3	
	3406h/C	VL.ARZF4	

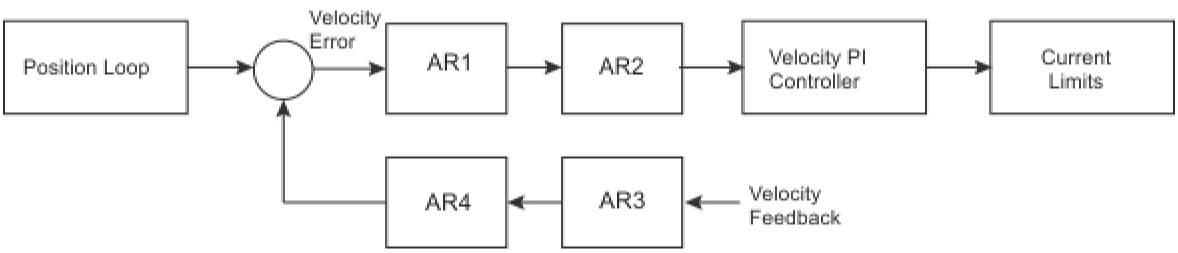
Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	832	VL.ARZF1	No	32 bit	No	M_01-03-00-000
	834	VL.ARZF2				
	836	VL.ARZF3				
	838	VL.ARZF4				

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:

$$ARx(\mathbf{s}) = [\mathbf{s}^2/(2\pi F_Z)^2 + \mathbf{s}/(Q_Z 2\pi F_Z) + 1]/ [\mathbf{s}^2/(2\pi F_P)^2 + \mathbf{s}/(Q_P 2\pi F_P) + 1]$$

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 ≈ $(1\text{-}z^{-1})/t$, where t = 62.5 µs

The poles are prewarped to F_P and the zeros are prewarped to F_Z.

Related Topics

Velocity Loop (pg 137)

25.41.5 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 (pg 921), VL.ARPQ1 TO VL.ARPQ4 (pg 923), VL.ARZF1 TO VL.ARZF4 (pg 926)
Start Version	M_01-02-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

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	

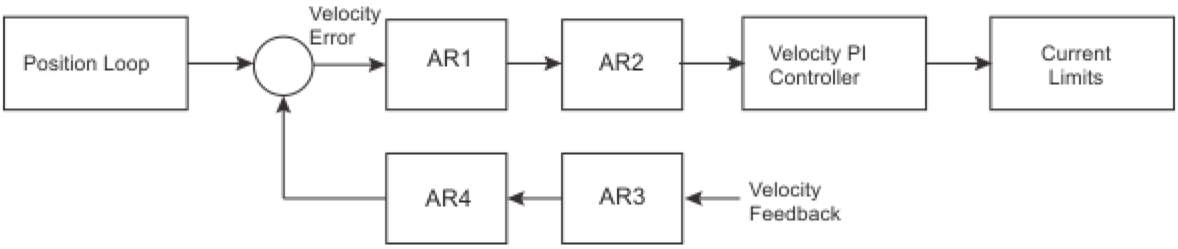
Fieldbus	Index/Subindex		Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	840	VL.ARZQ1	No	32 bit	No	M_01-03-00-000
	842	VL.ARZQ2				
	844	VL.ARZQ3				
	846	VL.ARZQ4				

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:

$$AR1(\mathbf{s}) = [\mathbf{s}^2/(2\pi F_Z)^2 + \mathbf{s}/(Q_Z 2\pi F_Z) + 1] / [\mathbf{s}^2/(2\pi F_P)^2 + \mathbf{s}/(Q_P 2\pi F_P) + 1]$$

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:

$$\mathbf{s} \approx (1-z^{-1})/t, \text{ where } t = 62.5\ \mu s.$$

The poles are prewarped to F_P and the zeros are prewarped to F_Z.

Related Topics

Velocity Loop (pg 137)

25.41.6 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 or Rotary: rpm, rps, deg/s, (custom units)/s, rad/s Linear: counts/s, mm/s, µm/s, (custom units)/s
Range	0.0 to VL.LIMITP (pg 949)
Default Value	0.0
Data Type	Float
See Also	VL.FF (pg 938), VL.KBUSFF (pg 940)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	60B1h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	848	Yes	low 32 bit word	Yes	M_01-03-00-000

Description

This parameter displays the velocity loop feedforward value injected by the fieldbus.

Related Topics

13.4 Velocity Loop

25.41.7 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 or 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 (pg 934), VL.CMDU (pg 932), VL.LIMITP (pg 949), VL.LIMITN (pg 947)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	606Bh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	850	Yes	low 32 bit word	Yes	M_01-03-00-000

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 (pg 947) and VL.LIMITP (pg 949)). See velocity loop design diagram for more details.

Related Topics

13.4 Velocity Loop
13.5 Position Loop
23 Block Diagrams

25.41.8 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 or Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: counts/s, mm/s, µm/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 µm/s -1,250.000 to 1,250.000 custom units/s
Default Value	0
Data Type	Float
See Also	VL.FB (pg 934), VL.CMD (pg 931), DRV.OPMODE (pg 549), DRV.CMDSOURCE (pg 497), VL.LIMITN (pg 947), VL.LIMITP (pg 949)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	60FFh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	852	Yes	low 32 bit word	Yes	M_01-03-00-000

Description

VL.CMDU sets the user velocity command. When DRV.OPMODE (pg 549) is set to 1 (velocity loop) and DRV.CMDSOURCE (pg 497) 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

13.4 Velocity Loop

25.41.9 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 or 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 (pg 931), VL.FB (pg 934)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3407h/4	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	854	Yes	low 32 bit word	Yes	M_01-03-00-000

Description

VL.ERR sets the velocity error. It is calculated in the velocity loop as the difference between VL.CMD (pg 931) and VL.FB (pg 934).

Related Topics

13.4 Velocity Loop

25.41.10 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 or 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 (pg 932)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3618h/0 606Ch/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	856	Yes	low 32 bit word	Yes	M_01-03-00-000

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

13.4 Velocity Loop

25.41.11 VL.FBFILTER

General Information	
Type	R/O Parameter
Description	Filters VL.FB (pg 934) 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, µm/s, (custom units)/s
Range	N/A
Default Value	N/A
Data Type	Float
See Also	VL.FB (pg 934)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3407h/1	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	858	Yes	low 32 bit word	Yes	M_01-03-00-000

Description

This parameter returns the same value as VL.FB (pg 934), filtered through a 10 Hz filter.

Related Topics

13.4 Velocity Loop

25.41.12 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 (pg 818)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	860	No	8 bit	No	M_01-03-00-000

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

Velocity Loop (pg 137)

25.41.13 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, µm/s, (custom units)/s
Range	N/A
Default Value	N/A
Data Type	Float
See Also	VL.FB (pg 934), VL.FBFILTER (pg 935)
Start Version	M_01-03-06-000

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1690	Yes	64-bit	Yes	M_01-06-03-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Description

VL.FBUNFILTERED reads the raw velocity feedback before any filters affect the value of this feedback.

Related Topics

Velocity Loop (pg 137)

25.41.14 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, µm/s, (custom units)/s
Range	0 to VL.LIMITP (pg 949)
Default Value	0
Data Type	Float
See Also	VL.KBUSFF (pg 940)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	862	Yes	low 32 bit word	Yes	M_01-03-00-000

Description

This parameter displays the velocity loop overall feedforward value.

Related Topics

13.4 Velocity Loop

VL.KVFF (pg 946)

25.41.15 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	864	No	16 bit	No	M_01-03-00-000

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

Velocity Loop (pg 137)

25.41.16 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 (pg 930)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3416h	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	866	No	32 bit	No	M_01-03-00-000

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 (pg 549) = 2).

Related Topics

13.4 Velocity Loop

25.41.17 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 (pg 944)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	354Dh/0	M_01-00-00-000

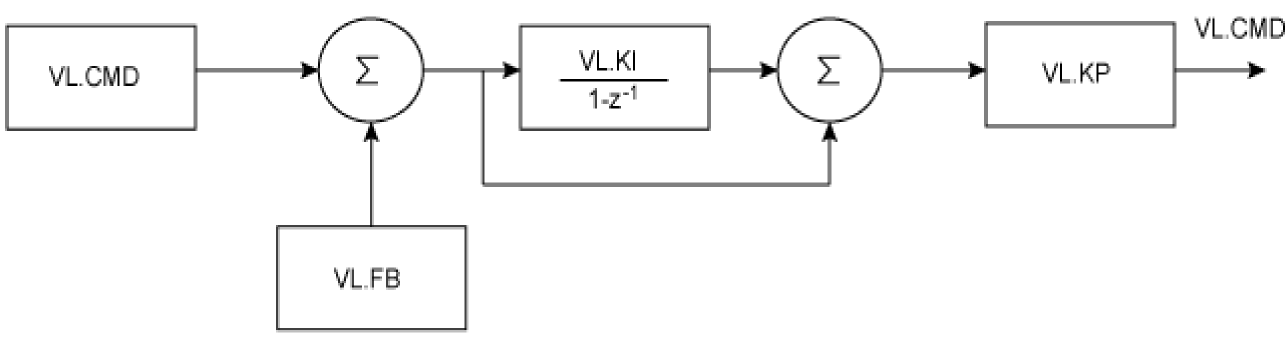
Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	868	No	32 bit	No	M_01-03-00-000

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 (pg 944) is set to 1, will take (1000/160)*2π 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

Velocity Loop (pg 137)

25.41.18 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 (pg 952)
Start Version	M_01-00-01-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	870	No	32 bit	No	M_01-03-00-000

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. (pg 952) will match VL.FB (pg 934), except when there is an unspecified offset between the two.

Related Topics

Velocity Loop (pg 137)

25.41.19 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 (pg 941)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3548h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	872	No	32 bit	No	M_01-03-00-000

Description

VL.KP sets the proportional gain of the velocity loop.

The idealized velocity loop bandwidth in Hz is:

Rotary motor:

Bandwidth (Hz) = VL.KP *K_t/ (2π *J_m)

Where:

K_t= motor torque constant, in units of Nm/Arms

J_m = motor inertia, in units of kg*m²

Linear motor:

Bandwidth (Hz) = VL.KP * K_t / (Motor Pitch (mm) * J_m)

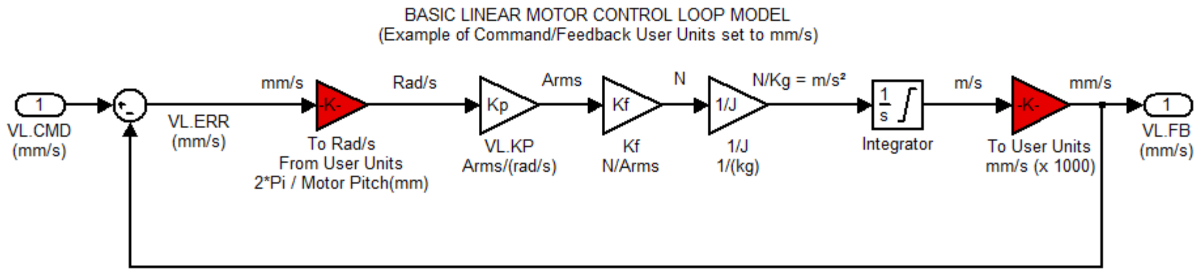
Where:

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 Arm /rad/s * (2π rad / 32mm MOTOR.PITCH)
VL.KP = 0.32 * 2π / 32 = 0.063 Arms / (mm/s)
```

Related Topics

Velocity Controller Environment Block Diagram

25.41.20 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 (pg 938)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3407h 354Bh/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	874	No	32 bit	No	M_01-03-00-000

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 (pg 549) = 2).

Related Topics

13.4 Velocity Loop

25.41.21 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 or Rotary: rpm, rps, deg/s, custom units/s, rad/s Linear: counts/s, mm/s, µm/s, custom units/s
Range	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 (pg 761) to 0.000 mm/s -250,000.000*MOTOR.PITCH (pg 761) to 0.000 µm/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 (pg 761) mm/s -50,000.004*MOTOR.PITCH µm/sec -250.000 custom units/s
Data Type	Float
See Also	VL.LIMITP (pg 949), VL.CMD (pg 931)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3623h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	876	Yes	low 32 bit word	Yes	M_01-03-00-000

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 (pg 931) is limited by the value of VL.LIMITN.

Related Topics

13.4 Velocity Loop

25.41.22 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 or 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 1570.796 rad/s Linear: 0.000 to 1,073,741,824,000.000 counts/s 0.000 to 250.000*MOTOR.PITCH (pg 761) mm/sec 0.000 to 250,000.000*MOTOR.PITCH (pg 761) µm/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 (pg 761) mm/sec 50,000.004*MOTOR.PITCH µm/sec 250.000 custom units/s
Data Type	Float
See Also	VL.LIMITN (pg 947), VL.CMD (pg 931)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3622h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	878	Yes	low 32 bit word	No	M_01-03-00-000

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 (pg 931) is limited by the value of VL.LIMITP.

Related Topics

13.4 Velocity Loop

25.41.23 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 (pg 693)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	880	No	32 bit	No	M_01-03-00-000

Description

This parameter is used in the internal calculation of the current loop acceleration feed forward gain value.

Related Topics

Velocity Loop (pg 137)

25.41.24 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, µm/s, (custom units)/s
Range	N/A
Default Value	N/A
Data Type	Float
See Also	VL.FB (pg 934), VL.KO (pg 943)
Start Version	M_01-00-01-000

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	882	Yes	low 32 bit word	Yes	M_01-03-00-000

Description

VL.MODEL is the observer velocity output. When VL.KO (pg 943) is tuned properly, VL.MODEL will match VL.FB (pg 934), except when there is an unspecified offset between the two.

Related Topics

13.4 Velocity Loop

25.41.25 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	884	No	32 bit	No	M_01-03-00-000

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

Velocity Loop (pg 137)

25.41.26 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

Variants Supported

Variant	Supported
AKD BASIC	N/A
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	886	No	32 bit	No	M_01-03-00-000

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

Velocity Loop (pg 137)

25.41.27 VL.THRESH

General Information	
Type	NV Parameter
Description	Sets the over speed fault 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, 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 250.000*MOTOR.PITCH (pg 761) mm/s 0.000 to 250,000.000*MOTOR.PITCHMOTOR.PITCH (pg 761)µm/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 (pg 761) mm/s 60,000.04*MOTOR.PITCHMOTOR.PITCH (pg 761)µm/s 300.000 custom units/s
Data Type	Float
See Also	VL.CMD (pg 931), VL.CMDU (pg 932)
Start Version	M_01-00-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3627h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	888	Yes	low 32 bit word	Yes	M_01-03-00-000

Description

VL.THRESH sets the threshold for the velocity over which an over speed fault is generated.

The value is considered as an absolute value, hence it applies for both negative and positive velocities.

Example

VL.THRESH is set to 600 rpm. A velocity (VL.FB (pg 934)) of 700 rpm will generate an over speed fault.

Related Topics

13.4 Velocity Loop

25.42 WS Parameters

This section describes the WS parameters.

25.42.1 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3494h/6	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	890	No	Command	No	M_01-03-00-000

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 (pg 70)

25.42.2 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 Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1734	No	8-bit	No

Description

AKD 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 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 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 (pg 70) | WS.MODE (pg 967)

25.42.3 WS.CHECKT

General Information	
Type	R/W Parameter
Description	Sets the amount of time a communication 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 Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1700	No	16-bit	No

Description

This parameter sets the amount of time a communication 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 (pg 70) | WS.CHECKV (pg 961)

25.42.4 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 Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1702	Yes	64-bit	Yes

Description

This parameter sets the velocity threshold which has to be exceeded to activate commutation monitoring.

Related Topics

Wake and Shake Overview (pg 70) | WS.CHECKT (pg 960)| UNIT.VROTARY (pg 910) | UNIT.VLINEAR (pg 909)

25.42.5 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	N/A

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	1694	No	Command	No	M_01-06-03-000

Description

This command disables wake and shake immediately. Feedback type is not relevant for this command. If WS.ARM (pg 958) has been issued, the request to execute the wake and shake algorithm at the next enable is cancelled. WS.STATE (pg 969) is set to IDLE.

Related Topics

Wake and Shake Overview (pg 70)

25.42.6 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3494h/2	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	892	Yes	64 bit	Yes	M_01-03-00-000

Description

This parameter sets the maximum movement that is allowed for finding commutation. If this value is too small, FF475 (pg 260), "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 (pg 70)

25.42.7 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	36D1h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	896	Yes	64 bit	Yes	M_01-03-00-000

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 (pg 70)

25.42.8 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

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1696	No	32-bit	No

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 (pg 967)

25.42.9 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 (pg 755), DRV.IPEAK (pg 535)
Start Version	M_01-01-00-101, M_01-02-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3494h/1	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	900	No	32 bit	Yes	M_01-03-00-000

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 (pg 70)

25.42.10 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
See Also	N/A
Start Version	M_01-01-00-101, M_01-02-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	902	No	8 bit	No	M_01-03-00-000

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 (pg 70)

25.42.11 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	36E2h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	904	No	8 bit	No	M_01-03-00-000

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 (pg 70)

25.42.12 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3494h/5	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	906	No	8 bit	No	M_01-03-00-000

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 (pg 70)

25.42.13 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 (pg 966), WS.DISTMAX
Start Version	M_01-01-00-101, M_01-02-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	36D0h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	908	No	16 bit	No	M_01-03-00-000

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 (pg 70)

25.42.14 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3683h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	910	No	16 bit	No	M_01-03-00-000

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 (pg 70)

25.42.15 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3685h/0	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	912	No	16 bit	No	M_01-03-00-000

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 (pg 70)

25.42.16 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

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CAN-open	3494h/3	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	914	No	16 bit	No	M_01-03-00-000

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 (pg 70)

25.42.17 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 Information

Fieldbus	Index/Subindex	Object Start Version
CANopen	0x535D/0	M_01-06-07-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1722	No	16-bit	No

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 (pg 70) | WS.MODE (pg 967) | WS.TSTANDSTILL (pg 975)

25.42.18 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 Information

Fieldbus	Index/Subindex	Object Start Version
CANopen	0x535C/0	M_01-06-07-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?
Modbus	1720	No	16-bit	No

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 (pg 70) | WS.MODE (pg 967) | WS.TIRAMP (pg 974)

25.42.19 WS.VTHRESH

General Information	
Type	NV Parameter
Description	Defines the maximum allowed velocity for Wake & Shake.
Units	Depends on UNIT.VROTARY (pg 910) or UNIT.VLINEAR (pg 909)UNIT.ACCLINEAR (pg 902) Rotary: rpm, rps, deg/s, Custom Units (pg 96)/s, rad/s Linear: counts/s, mm/s, µm/s, Custom Units (pg 96)/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 1,250.000 Custom Units (pg 96)/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 (pg 96)/s
Default Value	100 rpm
Data Type	U16
See Also	N/A
Start Version	M_01-01-00-101, M_01-02-00-000

Variants Supported

Variant	Supported
AKD BASIC	√
AKD SynqNet	√
AKD EtherNet/IP	√

Fieldbus Information

Fieldbus	Index/Subindex	Object Start Version
EtherCAT COE and CANopen	3494h/4	M_01-00-00-000

Fieldbus	Index/Subindex	Is 64 bit?	Attributes	Signed?	Object Start Version
Modbus	916	Yes	low 32 bit word	Yes	M_01-03-00-000

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 F478 (pg 260) 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 (pg 70)

26 Index

A

B

C

D

E

O

P

R

S

T

U

V

W

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.

For assistance with your application needs, visit www.kollmorgen.com or contact us at:

North America
KOLLMORGEN
203A West Rock Road
Radford, VA 24141 USA

Internet	www.kollmorgen.com
E-Mail	support@kollmorgen.com
Tel.:	+1 - 540 - 633 - 3545
Fax:	+1 - 540 - 639 - 4162

Europe
KOLLMORGEN Europe GmbH
Pempelfurtstraße 1
40880 Ratingen, Germany

Internet	www.kollmorgen.com
E-Mail	technik@kollmorgen.com
Tel.:	+49 - 2102 - 9394 - 0
Fax:	+49 - 2102 - 9394 - 3155

Asia
KOLLMORGEN
Rm 2205, Scitech Tower, China
22 Jianguomen Wai Street

Internet	www.kollmorgen.com
E-Mail	sales.asia@kollmorgen.com
Tel.:	+86 - 400 666 1802
Fax:	+86 - 10 6515 0263

KOLLMORGEN®

Because Motion Matters™