

ACS850

Firmware Manual
ACS850 Standard Control Program

efesotomasyon.com



ACS850 drive manuals

efesotomasyon.com

DRIVE HARDWARE MANUAL*

ACS850-04 Drive Modules (0.37 to 45 kW) Hardware Manual – 3AUA0000045496 (English)

ACS850-04 Drive Modules (55 to 160 kW, 75 to 200 hp) Hardware Manual – 3AUA0000045487 (English)

ACS850-04 Drive Modules (200 to 500 kW, 250 to 600 hp) Hardware Manual – 3AUA0000026234 (English)

DRIVE FIRMWARE MANUALS

ACS850 Standard Control Program Firmware Manual** – 3AUA0000045497 (English)

OPTION MANUALS

ACS-CP-U Control Panel IP54 Mounting Platform Kit (+J410) Installation Guide – 3AUA0000049072 (English)

Manuals for I/O Extension Modules, Fieldbus Adapters, etc.*

APPLICATION GUIDES

Application programming for ACS850 drives – 3AUA0000078664 (English)

ATEX certified safe disconnection function for ACS850 drives – 3AUA0000074343 (English)

Safe torque off function for ACSM1, ACS850 and ACQ810 drives – 3AFE68929814 (English)

*The delivery includes a multilingual quick installation guide.

**The delivery includes a multilingual quick start-up guide.

Firmware Manual

ACS850 Standard Control Program

efesotomasyon.com

Table of contents



Table of contents

ACS850 drive manuals	2
----------------------------	---

1. About the manual

What this chapter contains	11
Compatibility	11
Safety instructions	11
Reader	11
Contents	12
Related manuals	12
Terms and abbreviations	12

2. The ACS850 control panel

What this chapter contains	15
Features	15
Installation	16
Mechanical installation	16
Electrical installation	16
Layout	17
Status line	18
Operating instructions	19
Basics of operation	19
List of tasks	20
Help and panel version – Any mode	21
Basic operations – Any mode	22
Output mode	23
Parameters	25
Assistants	32
Changed Parameters	33
Fault Logger	35
Time & Date	37
Parameter Backup	39
I/O Settings	47
Reference Edit	49
Drive Info	50
Parameter Change Log	51

3. Control locations and operating modes

What this chapter contains	53
Local control vs. external control	54
Local control	54
External control	55
Operating modes of the drive	55
Speed control mode	55
Torque control mode	55



Special control modes	55
-----------------------------	----

4. Program features

What this chapter contains	57
Drive configuration and programming	57
Programming via parameters	58
Application programming	58
Control interfaces	59
Programmable analog inputs	59
Programmable analog outputs	59
Programmable digital inputs and outputs	59
Programmable I/O extensions	60
Programmable relay outputs	60
Fieldbus control	60
Motor control	61
Constant speeds	61
Critical speeds	61
Speed controller tuning	61
Encoder support	63
Jogging	64
Scalar motor control	65
User-definable load curve	66
User-definable U/f curve	67
Autophasing	67
Flux braking	69
Application control	70
Application macros	70
Process PID control	70
Mechanical brake control	72
Timers	76
DC voltage control	77
Overvoltage control	77
Undervoltage control	77
Voltage control and trip limits	78
Brake chopper	79
Safety and protections	80
Emergency stop	80
Thermal motor protection	80
Programmable protection functions	83
Automatic fault resets	84
Diagnostics	84
Signal supervision	84
Maintenance counters	84
Energy saving calculator	85
Load analyzer	85
Miscellaneous	86
Backup and restore of drive contents	86
Data storage parameters	88
Drive-to-drive link	88



5. Application macros

What this chapter contains	89
General	89
Factory macro	90
Default control connections for the Factory macro	91
Hand/Auto macro	92
Default control connections for the Hand/Auto macro	93
PID control macro	94
Default control connections for the PID control macro	95
Torque control macro	96
Default control connections for the Torque control macro	97
Sequential control macro	98
Default control connections for the Sequential control macro	100

6. Parameters

What this chapter contains	101
Terms and abbreviations	102
Parameter listing	103
01 Actual values	103
02 I/O values	104
03 Control values	114
04 Appl values	115
06 Drive status	116
08 Alarms & faults	119
09 System info	121
10 Start/stop/dir	122
11 Start/stop mode	129
12 Operating mode	131
13 Analogue inputs	133
14 Digital I/O	140
15 Analogue outputs	153
16 System	160
19 Speed calculation	162
20 Limits	166
21 Speed ref	168
22 Speed ref ramp	170
23 Speed ctrl	173
24 Torque ref	181
25 Critical speed	182
26 Constant speeds	183
27 Process PID	185
30 Fault functions	189
31 Motor therm prot	192
32 Automatic reset	197
33 Supervision	198
34 User load curve	201
35 Process variable	203
36 Timed functions	209
38 Flux ref	214



8 Table of contents

40 Motor control	215
42 Mech brake ctrl	217
44 Maintenance	221
45 Energy optimising	227
47 Voltage ctrl	228
48 Brake chopper	228
49 Data storage	229
50 Fieldbus	230
51 FBA settings	232
52 FBA data in	233
53 FBA data out	234
56 Panel display	234
57 D2D communication	235
58 Embedded Modbus	237
64 Load analyzer	241
90 Enc module sel	244
91 Absol enc conf	246
92 Resolver conf	248
93 Pulse enc conf	249
94 Ext IO conf	250
95 Hw configuration	250
97 User motor par	251
99 Start-up data	252

7. Additional parameter data

What this chapter contains	257
Terms and abbreviations	257
Fieldbus equivalentents	258
Pointer parameter format in fieldbus communication	258
32-bit integer value pointers	258
32-bit integer bit pointers	259
Parameter groups 1...9	261
Parameter groups 10...99	264

8. Fault tracing

What this chapter contains	283
Safety	283
How to reset	283
Fault history	284
Alarm messages generated by the drive	284
Fault messages generated by the drive	292

9. Control through the embedded fieldbus interface

What this chapter contains	303
System overview	304
Connecting the embedded fieldbus to the drive	305
Setting up the embedded fieldbus interface	306
Setting the drive control parameters	308

Basics of the embedded fieldbus interface	310
Control Word and Status Word	311
References	311
Actual values	311
Data inputs/outputs	311
About the EFB communication profiles	312
ABB Drives classic profile and ABB Drives enhanced profile	313
Control Word for the ABB Drives profiles	313
Status Word for the ABB Drives profiles	315
State transition diagram for the ABB Drives profiles	317
References for the ABB Drives profiles	318
Actual values for the ABB Drives profiles	319
Modbus register addresses for the ABB Drives classic profile	320
Modbus register addresses for the ABB Drives enhanced profile	321
DCU 16-bit profile	322
Control and Status words for the DCU 16-bit profile	322
Status Word for the DCU 16-bit profile	322
State transition diagram for the DCU 16-bit profile	322
References for the DCU 16-bit profile	322
Actual signals for the DCU 16-bit profile	322
Modbus register addresses for the DCU 16-bit profile	323
DCU 32-bit profile	324
Control and Status words for the DCU 32-bit profile	324
Status word for the DCU 32-bit profile	324
State transition diagram for the DCU 32-bit profile	324
References for the DCU 32-bit profile	325
Actual signals for the DCU 32-bit profile	326
Modbus register addresses for the DCU 32-bit profile	327
Modbus function codes	328
Modbus exception codes	329

10. Control through a fieldbus adapter

What this chapter contains	331
System overview	332
Setting up communication through a fieldbus adapter module	333
Setting the drive control parameters	335
Basics of the fieldbus adapter interface	336
Control Word and Status Word	337
Actual values	337
FBA communication profile	337
Fieldbus references	338
State diagram	339

11. Drive-to-drive link

What this chapter contains	341
General	341
Wiring	341
Datasets	342
Types of messaging	343



10 Table of contents

Master point-to-point messaging	343
Read remote messaging	344
Follower point-to-point messaging	344
Standard multicast messaging	345
Broadcast messaging	346
Chained multicast messaging	348

12. Control chain and drive logic diagrams

What this chapter contains	351
Speed feedback	352
Speed reference modification and ramping	353
Speed error handling	354
Torque reference modification, operating mode selection	355
Process PID	356
Drive logic 1	357
Drive logic 2 (Fieldbus interface)	358
Direct torque control	359



Further information

Product and service inquiries	361
Product training	361
Providing feedback on ABB Drives manuals	361
Document library on the Internet	361

efesotomasyon.com

1

About the manual

What this chapter contains

The chapter describes the contents of the manual. It also contains information on the compatibility, safety and intended audience.

Compatibility

The manual is compatible with ACS850 standard control program version UIF12100 and later.

Safety instructions

Follow all safety instructions delivered with the drive.

- Read the **complete safety instructions** before you install, commission, or use the drive. The complete safety instructions are given at the beginning of the *Hardware Manual*.
- Read the **software function specific warnings and notes** before changing the default settings of the function. For each function, the warnings and notes are given in this manual in the section describing the related user-adjustable parameters.

Reader

The reader of the manual is expected to know the standard electrical wiring practices, electronic components, and electrical schematic symbols.

Contents

The manual consists of the following chapters:

- [The ACS850 control panel](#) provides a description and instructions for use of the control panel.
- [Control locations and operating modes](#) describes the control locations and operation modes of the drive.
- [Program features](#) contains descriptions of the features of the ACS850 standard control program.
- [Application macros](#) contains a short description of each macro together with a connection diagram.
- [Parameters](#) describes the parameters of the drive.
- [Additional parameter data](#) contains further information on the parameters.
- [Fault tracing](#) lists the alarm (warning) and fault messages with possible causes and remedies.
- [Control through the embedded fieldbus interface](#) describes the communication to and from a fieldbus network using an embedded fieldbus interface.
- [Control through a fieldbus adapter](#) describes the communication to and from a fieldbus network using an optional fieldbus adapter module.
- [Drive-to-drive link](#) describes the communication between drives connected together by the drive-to-drive link.
- [Control chain and drive logic diagrams](#).

Related manuals

The delivery of the drive includes a multilingual *Quick Start-up Guide*.

A complete list of related manuals is printed on the inside of the front cover.

Terms and abbreviations

Term/abbreviation	Definition
AI	Analog input; interface for analog input signals
AO	Analog output; interface for analog output signals
DC link	DC circuit between rectifier and inverter
DI	Digital input; interface for digital input signals
DO	Digital output; interface for digital output signals
DTC	Direct torque control
EFB	Embedded fieldbus
FBA	Fieldbus adapter
FEN-01	Optional TTL encoder interface module for the ACS850
FEN-11	Optional absolute encoder interface module for the ACS850
FEN-21	Optional resolver interface module for the ACS850

Term/abbreviation	Definition
FEN-31	Optional HTL encoder interface module for the ACS850
FIO-01	Optional digital I/O extension module for the ACS850
FIO-11	Optional analog I/O extension module for the ACS850
FIO-21	Optional analog/digital I/O extension module for the ACS850
FCAN-0x	Optional CANopen adapter for the ACS850
FDNA-0x	Optional DeviceNet adapter for the ACS850
FECA-01	Optional EtherCAT® adapter for the ACS850
FENA-0x	Optional Ethernet/IP adapter for the ACS850
FLON-0x	Optional LONWORKS® adapter for the ACS850
FPBA-0x	Optional PROFIBUS DP adapter for the ACS850
FSCA-0x	Optional Modbus adapter for the ACS850
HTL	High-threshold logic
IGBT	Insulated gate bipolar transistor; a voltage-controlled semiconductor type widely used in inverters due to their easy controllability and high switching frequency
I/O	Input/Output
ID run	Motor identification run. During the identification run, the drive will identify the characteristics of the motor for optimum motor control.
JCU	Control unit of the drive module. The JCU is installed on top of the power unit. The external I/O control signals are connected to the JCU, or optional I/O extensions mounted on it.
JMU	Memory unit attached to the control unit of the drive
JPU	<i>Power unit</i> ; see the definition below.
LSB	Least significant bit
LSW	Least significant word
MSB	Most significant bit
MSW	Most significant word
Parameter	User-adjustable operation instruction to the drive, or signal measured or calculated by the drive
PI controller	Proportional-integral controller
PID controller	Proportional–integral–derivative controller. Drive speed control is based on PID algorithm.
PLC	Programmable logic controller
Power unit	Contains the power electronics and connections of the drive module. The JCU is connected to the power unit.
PTC	Positive temperature coefficient
RFG	Ramp Function Generator
RO	Relay output; interface for a digital output signal. Implemented with a relay.
SSI	Synchronous serial interface
STO	Safe torque off
TTL	Transistor-transistor logic
UIFI xxxx	Firmware of the ACS850 drive

14 About the manual

Term/abbreviation	Definition
UPS	Uninterruptible power supply; power supply equipment with battery to maintain output voltage during power failure

efesotomasyon.com



2

The ACS850 control panel

What this chapter contains

This chapter describes the features and operation of the ACS850 control panel.

The control panel can be used to control the drive, read status data, and adjust parameters.

Features

- alphanumeric control panel with an LCD display
 - copy function – parameters can be copied to the control panel memory for later transfer to other drives or for backup of a particular system.
 - context sensitive help
 - real time clock.
-

Installation

■ Mechanical installation

For mounting options, see the *Hardware Manual* of the drive.

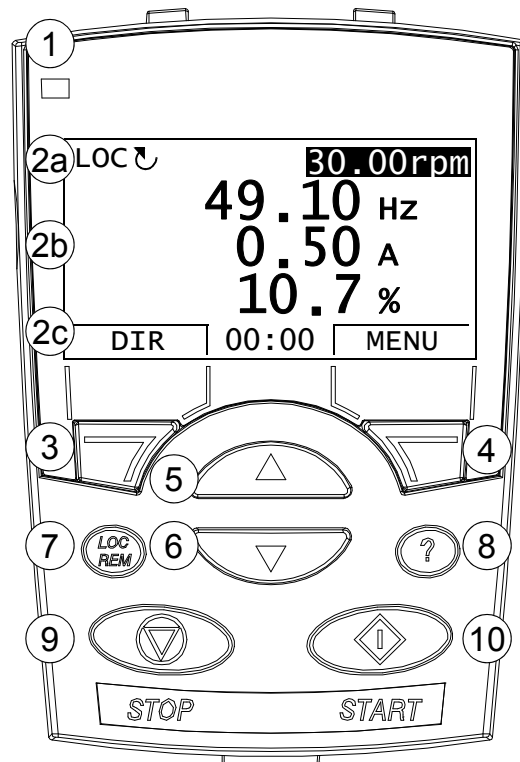
Instructions for mounting the control panel onto a cabinet door are available in *ACS-CP-U Control Panel IP54 Mounting Platform Kit Installation Guide* (3AUA0000049072 [English]).

■ Electrical installation

Use a CAT5 straight-through network cable with a maximum length of 3 meters. Suitable cables are available from ABB.

For the control panel connector location on the drive, see the *Hardware Manual* of the drive.

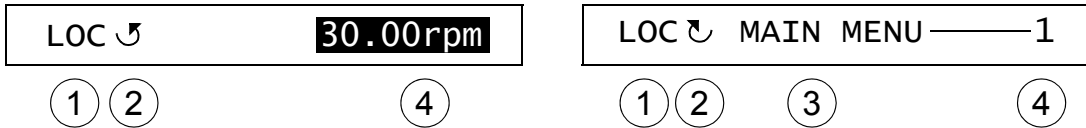
Layout



No.	Use
1	Status LED – Green for normal operation.
2	LCD display – Divided into three main areas: Status line – variable, depending on the mode of operation, see section Status line on page 18. Center – variable; in general, shows signal and parameter values, menus or lists. Shows also faults and alarms. Bottom line – shows current functions of the two soft keys and, if enabled, the clock display.
3	Soft key 1 – Function depends on the context. The text in the lower left corner of the LCD display indicates the function.
4	Soft key 2 – Function depends on the context. The text in the lower right corner of the LCD display indicates the function.
5	Up – Scrolls up through a menu or list displayed in the center of the LCD display. Increments a value if a parameter is selected. Increments the reference value if the upper right corner is highlighted. Holding the key down changes the value faster.
6	Down – Scrolls down through a menu or list displayed in the center of the LCD display. Decrements a value if a parameter is selected. Decrements the reference value if the upper right corner is highlighted. Holding the key down changes the value faster.
7	LOC/REM – Changes between local and remote control of the drive.
8	Help – Displays context sensitive information when the key is pressed. The information displayed describes the item currently highlighted in the center of the display.
9	STOP – Stops the drive in local control.
10	START – Starts the drive in local control.

■ Status line

The top line of the LCD display shows the basic status information of the drive.





No.	Field	Alternatives	Significance
1	Control location	LOC	Drive control is local, that is, from the control panel.
		REM	Drive control is remote, such as the drive I/O or fieldbus.
2	State	↻	Forward shaft direction
		↴	Reverse shaft direction
		Rotating arrow	Drive is running at reference.
		Dotted rotating arrow	Drive is running but not at reference.
		Stationary arrow	Drive is stopped.
		Dotted stationary arrow	Start command is present, but the motor is not running, e.g. because start enable signal is missing.
3	Panel operation mode		<ul style="list-style-type: none"> • Name of the current mode • Name of the list or menu shown • Name of the operation state, e.g. REF EDIT.
4	Reference value or number of the selected item		<ul style="list-style-type: none"> • Reference value in the Output mode • Number of the highlighted item, e.g mode, parameter group or fault.

Operating instructions

■ Basics of operation

You operate the control panel with menus and keys. The keys include two context-sensitive soft keys, whose current function is indicated by the text shown in the display above each key.

You select an option, e.g. operation mode or parameter, by entering the MENU state using soft key 2, and then by scrolling the  and  arrow keys until the option is highlighted and then pressing the relevant soft key. With the right soft key you usually enter a mode, accept an option or save the changes. The left soft key is used to cancel the made changes and return to the previous operation level.

The Control Panel has ten options in the Main menu: Parameters, Assistants, Changed Par, Fault Logger, Time & Date, Parameter Backup, I/O Settings, Reference Edit, Drive Info and Parameter Change Log. In addition, the control panel has an Output mode, which is used as default. Also, when a fault or alarm occurs, the panel goes automatically to the Fault mode showing the fault or alarm. You can reset the fault in the Output or Fault mode. The operation in these modes and options is described in this chapter.

Initially, the panel is in the Output mode, where you can start, stop, change the direction, switch between local and remote control, modify the reference value and monitor up to three actual values. To do other tasks, go first to the Main menu and select the appropriate option on the menu. The status line (see section [Status line](#) on page 18) shows the name of the current menu, mode, item or state.

LOC ↻	30.00rpm
49.10	Hz
0.50	A
10.7	%
DIR	MENU

LOC ↻	MAIN MENU	1
PARAMETERS		
ASSISTANTS		
CHANGED PAR		
EXIT	ENTER	

■ List of tasks





The table below lists common tasks, the mode in which you can perform them, abbreviations of the options in the Main menu and the page number where the steps to do the task are described in detail.

Task	Mode / Main menu option	Abbreviations of the Main menu options *	Page
How to get help	Any	-	21
How to find out the panel version	Any	-	21
How to start and stop the drive	Output	-	22
How to switch between local and remote control	Any	-	22
How to change the direction of the motor rotation	Any	-	23
How to set the speed, frequency or torque reference in the Output mode	Output	-	23
How to adjust the display contrast	Output	-	24
How to change the value of a parameter	Parameters	PARAMETERS	25
How to change the value of value pointer parameters	Parameters	PARAMETERS	26
How to change the value of bit pointer parameter to point to the value of a bit in another signal	Parameters	PARAMETERS	28
How to change the value of bit pointer parameter to fixed 0 (FALSE) or 1 (TRUE)	Parameters	PARAMETERS	30
How to select the monitored signals	Parameters	PARAMETERS	31
How to do guided tasks (specification of related parameter sets) with assistants	Assistants	ASSISTANTS	32
How to view and edit changed parameters	Changed Parameters	CHANGED PAR	33
How to view faults	Fault Logger	FAULT LOGGER	35
How to reset faults and alarms	Fault Logger	FAULT LOGGER	36
How to show/hide the clock, change date and time formats, set the clock and enable/disable automatic clock transitions according to the daylight saving changes	Time & Date	TIME & DATE	37
How to copy parameters from the drive to the control panel	Parameter Backup	PAR BACKUP	39
How to restore parameters from the control panel to the drive	Parameter Backup	PAR BACKUP	39
How to view backup information	Parameter Backup	PAR BACKUP	45
How to edit and change parameter settings related to I/O terminals	I/O Settings	I/O SETTINGS	47
How to edit reference value	Reference Edit	REF EDIT	49
How to view drive info	Drive Info	DRIVE INFO	50
How to view and edit recently changed parameters	Parameter Change Log	PAR CHG LOG	51



* Main menu options actually shown in the control panel.

■ Help and panel version – Any mode

How to get help

Step	Action	Display
1.	<p>Press  to read the context-sensitive help text for the item that is highlighted.</p> <p>If help text exists for the item, it is shown on the display.</p>	<pre> LOC ↶ TIME & DATE — 6 TIME FORMAT DATE FORMAT SET TIME SET DATE DAYLIGHT SAVING EXIT 00:00 SEL </pre> <pre> LOC ↶ HELP — Use daylight saving to enable or disable automatic clock adjustment according to daylight saving EXIT 00:00 </pre>
2.	<p>If the whole text is not visible, scroll the lines with keys  and .</p>	<pre> LOC ↶ HELP — to enable or disable automatic clock adjustment according to daylight saving changes EXIT 00:00 </pre>
3.	<p>After reading the text, return to the previous display by pressing .</p>	<pre> LOC ↶ TIME & DATE — 6 TIME FORMAT DATE FORMAT SET TIME SET DATE DAYLIGHT SAVING EXIT 00:00 SEL </pre>










How to find out the panel version

Step	Action	Display
1.	<p>If the power is switched on, switch it off.</p> <ul style="list-style-type: none"> - If the panel cable can be disconnected easily, unplug the panel cable from the control panel, OR - if the panel cable can not be disconnected easily, switch off the control board or the drive. 	
2.	<p>Keep key  depressed while you switch on the power and read the information. The display shows the following panel information:</p> <p>Panel SW: Panel firmware version ROM CRC: Panel ROM check sum Flash Rev: Flash content version Flash content comment.</p> <p>When you release the  key, the panel goes to the Output mode.</p>	<pre> PANEL VERSION INFO Panel SW: x.xx Rom CRC: xxxxxxxxxx Flash Rev: x.xx XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX </pre>

■ Basic operations – Any mode

How to start, stop and switch between local and remote control


You can start, stop and switch between local and remote control in any mode. To be able to start or stop the drive by using the control panel, the drive must be in local control.

Step	Action	Display
1.	<p>To switch between remote control (REM shown on the status line) and local control (LOC shown on the status line), press .</p> <p>Note: Switching to local control can be prevented with parameter 16.01 Local lock.</p> <p>The very first time the drive is powered up, it is in remote control (REM) and controlled through the drive I/O terminals. To switch to local control (LOC) and control the drive using the control panel, press . The result depends on how long you press the key: If you release the key immediately (the display flashes “Switching to the local control mode”), the drive stops. Set the local control reference as instructed on page 23. If you press the key until the text “Keep running” appears, the drive continues running as before. The drive copies the current remote values for the run/stop status and the reference, and uses them as the initial local control settings.</p> <p>To stop the drive in local control, press .</p> <p>To start the drive in local control, press .</p>	<div style="border: 1px solid black; padding: 5px;"> <p>LOC  MESSAGE _____</p> <p>Switching to the local control mode.</p> <hr/> <p style="text-align: center;">00:00</p> </div> <p>The arrow ( or ) on the status line stops rotating.</p> <p>The arrow ( or ) on the status line starts rotating. It is dotted until the drive reaches the setpoint.</p>

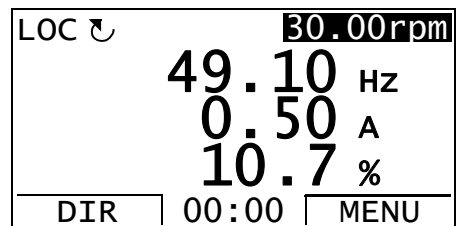
■ Output mode

In the Output mode, you can:




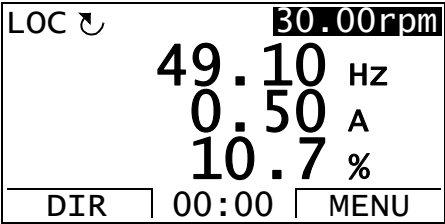



- monitor actual values of up to three signals
- change the direction of the motor rotation
- set the speed, frequency or torque reference
- adjust the display contrast
- start, stop, change the direction and switch between local and remote control.

You get to the Output mode by pressing  repeatedly.

The top right corner of the display shows the reference value. The center can be configured to show up to three signal values or bar graphs; see page 31 for instructions on selecting and modifying the monitored signals.


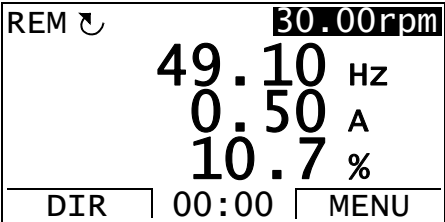


How to change the direction of the motor rotation


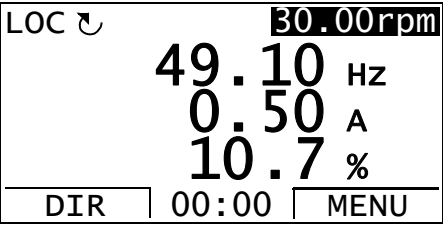


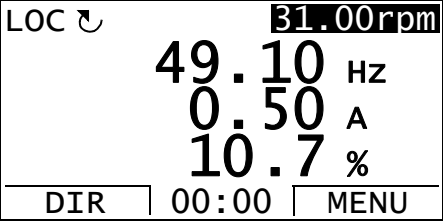
Step	Action	Display
1.	If you are not in the Output mode, press  repeatedly until you get there.	
2.	If the drive is in remote control (REM shown on the status line), switch to local control by pressing  . The display briefly shows a message about changing the mode and then returns to the Output mode.	
3.	To change the direction from forward ( shown on the status line) to reverse ( shown on the status line), or vice versa, press  .	

How to set the speed, frequency or torque reference in the Output mode


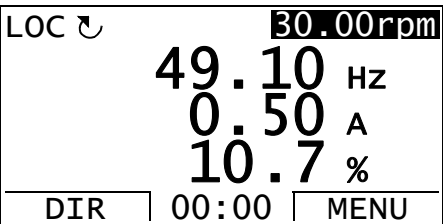




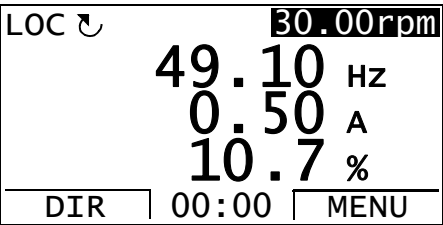
See also section [Reference Edit](#) on page 49.

Step	Action	Display
1.	If you are not in the Output mode, press  repeatedly until you get there.	

24 The ACS850 control panel

Step	Action	Display
2.	If the drive is in remote control (REM shown on the status line), switch to local control by pressing  . The display briefly shows a message about changing the mode and then returns to the Output mode.	 <p>LOC ↻ 30.00rpm 49.10 Hz 0.50 A 10.7 % DIR 00:00 MENU</p>
3.	To increase the highlighted reference value shown in the top right corner of the display, press  . The value changes immediately. It is stored in the permanent memory of the drive and restored automatically after power switch-off. To decrease the value, press  .	 <p>LOC ↻ 31.00rpm 49.10 Hz 0.50 A 10.7 % DIR 00:00 MENU</p>

How to adjust the display contrast












Step	Action	Display
1.	If you are not in the Output mode, press  repeatedly until you get there.	 <p>LOC ↻ 30.00rpm 49.10 Hz 0.50 A 10.7 % DIR 00:00 MENU</p>
2.	To increase the contrast, press keys  and  simultaneously. To decrease the contrast, press keys  and  simultaneously.	 <p>LOC ↻ 30.00rpm 49.10 Hz 0.50 A 10.7 % DIR 00:00 MENU</p>





■ Parameters

In the Parameters option, you can:

- view and change parameter values
- start, stop, change the direction and switch between local and remote control.








How to select a parameter and change its value













Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	<pre> LOC ↻ MAIN MENU — 1 PARAMETERS ASSISTANTS CHANGED PAR EXIT 00:00 ENTER </pre>
2.	Go to the Parameters option by selecting PARAMETERS on the menu with keys  and  , and pressing  .	<pre> LOC ↻ PAR GROUPS — 01 01 Actual values 02 I/O values 03 Control values 04 Appl values 06 Drive status EXIT 00:00 SEL </pre>
3.	Select the appropriate parameter group with keys  and  .	<pre> LOC ↻ PAR GROUPS — 99 99 Start-up data 01 Actual values 02 I/O values 03 Control values 04 Appl values EXIT 00:00 SEL </pre>
	Press  .	<pre> LOC ↻ PARAMETERS — 9901 Language English 9904 Motor type 9905 Motor ctrl mode 9906 Mot nom current EXIT 00:00 EDIT </pre>
4.	Select the appropriate parameter with keys  and  . The current value of the parameter is shown below the selected parameter. Here the parameter 99.06 <i>Mot nom current</i> is used as an example.	<pre> LOC ↻ PARAMETERS — 9901 Language 9904 Motor type 9905 Motor ctrl mode 9906 Mot nom current 0.0 A EXIT 00:00 EDIT </pre>
	Press  .	<pre> LOC ↻ PAR EDIT — 9906 Mot nom current 0.0 A CANCEL 00:00 SAVE </pre>

Step	Action	Display
5.	Specify a new value for the parameter with keys  and  . Pressing an arrow key once increments or decrements the value. Keeping the key depressed for a while first quickly changes the current digit until the cursor moves left one position. This is repeated until the key is released. After the key is released, step-by-step adjustment of the current digit is possible. If neither key is pressed for a while, the cursor returns to the right one position at a time. Pressing both keys simultaneously replaces the displayed value with the default value.	<pre> LOC ↺ PAR EDIT 9906 Mot nom current 3.5 A ----- CANCEL 00:00 SAVE </pre>
6.	To save the new value, press  . To cancel the new value and keep the original, press  .	<pre> LOC ↺ PARAMETERS 9906 Mot nom current 3.5 A 9907 Mot nom voltage 9908 Mot nom freq 9909 Mot nom speed ----- EXIT 00:00 EDIT </pre>

How to change the value of value pointer parameters












In addition to the parameters shown above, there are two kinds of pointer parameters; value pointer parameters and bit pointer parameters. A value pointer parameter points to the value of another parameter.










Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	<pre> LOC ↺ MAIN MENU ——— 1 PARAMETERS ASSISTANTS CHANGED PAR ----- EXIT 00:00 ENTER </pre>
2.	Go to the Parameters option by selecting PARAMETERS on the menu with keys  and  , and pressing  .	<pre> LOC ↺ PAR GROUPS ——— 01 01 Actual values 02 I/O values 03 Control values 04 Appl values 06 Drive status ----- EXIT 00:00 SEL </pre>
3.	Select the appropriate parameter group with keys  and  . Here the value pointer parameter 21.01 Speed ref1 sel is used as an example.	<pre> LOC ↺ PAR GROUPS ——— 21 15 Analogue outputs 16 System 19 Speed calculation 20 Limits 21 Speed ref ----- EXIT 00:00 SEL </pre>

Step	Action	Display
4.	Press  to select the appropriate parameter group. Select the appropriate parameter with keys  and  , current value of each parameter is shown below it.	<pre> LOC ↶ PARAMETERS 2101 Speed ref1 sel AI2 scaled 2102 Speed ref2 sel 2103 Speed ref1 func 2104 Speed ref1/2 sel EXIT 00:00 EDIT </pre>
5.	Press  . Current value of the value pointer parameter is shown, as well as the parameter it points to.	<pre> LOC ↶ PAR EDIT 2101 Speed ref1 sel AI1 scaled [P.02.05] CANCEL 00:00 SEL </pre>
6.	Specify a new value with keys  and  . The parameter the value pointer parameter points to changes respectively.	<pre> LOC ↶ PAR EDIT 2101 Speed ref1 sel FBA ref1 [P.02.26] CANCEL 00:00 SEL </pre>
7.	<p>Press  to accept any of the preselected values and to return to the parameters list. The new value is shown in the parameters list.</p> <p>To freely define an analog signal as the value, choose Pointer and press . The parameter group and index will be shown. Select the parameter group with  and . The text below the cursor displays the currently-selected parameter group.</p>	<pre> LOC ↶ PARAMETERS 2101 Speed ref1 sel FBA ref1 2102 Speed ref2 sel 2103 Speed ref1 func 2104 Speed ref1/2 sel EXIT 00:00 EDIT </pre> <pre> LOC ↶ PAR EDIT 2101 Speed ref1 sel P.02.05 02 I/O values CANCEL 00:00 SAVE </pre>
8.	Press  to select the parameter index. Again, the text below the cursor reflects the current setting.	<pre> LOC ↶ PAR EDIT 2101 Speed ref1 sel P.02.07 0207 AI2 scaled CANCEL 00:00 SAVE </pre>
9.	To save the new value for the pointer parameter, press  . The new value is shown in the parameters list.	<pre> LOC ↶ PARAMETERS 2101 Speed ref1 sel AI2 scaled 2102 Speed ref2 sel 2103 Speed ref1 func 2104 Speed ref1/2 sel EXIT 00:00 EDIT </pre>

How to change the value of bit pointer parameters

The bit pointer parameter points to the value of a bit in another signal, or can be fixed to 0 (FALSE) or 1 (TRUE). For the latter option, see page 30. A bit pointer parameter points to a bit value (0 or 1) of one bit in a 32-bit signal. The first bit from the left is bit number 31, and the first bit from the right is bit number 0.














Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	<pre> LOC ↺ MAIN MENU——1 PARAMETERS ASSISTANTS CHANGED PAR EXIT 00:00 ENTER </pre>
2.	Go to the Parameters option by selecting PARAMETERS on the menu with keys  and  , and pressing  .	<pre> LOC ↺ PAR GROUPS——01 01 Actual values 02 I/O values 03 Control values 04 Appl values 06 Drive status EXIT 00:00 SEL </pre>
3.	Select the appropriate parameter group with keys  and  . Here the bit pointer parameter 10.02 Ext1 start in1 is used as an example.	<pre> LOC ↺ PAR GROUPS——10 10 Start/stop/dir 11 Start/stop mode 12 Operating mode 13 Analogue inputs 14 Digital I/O EXIT 00:00 SEL </pre>
4.	Press  to select the appropriate parameter group. Current value of each parameter is shown below its name. Select the parameter 10.02 Ext1 start in1 with keys  and  .	<pre> LOC ↺ PARAMETERS—— 1001 Ext1 start func In1 1002 Ext1 start in1 1003 Ext1 start in2 1004 Ext2 start func EXIT 00:00 EDIT </pre> <pre> LOC ↺ PARAMETERS—— 1001 Ext1 start func 1002 Ext1 start in1 DI1 1003 Ext1 start in2 1004 Ext2 start func EXIT 00:00 EDIT </pre>
5.	Press  .	<pre> LOC ↺ PAR EDIT—— 1002 Ext1 start in1 DI1 [P.02.01.00] CANCEL 00:00 SEL </pre>






Step	Action	Display
6.	Specify a new value with keys  and  . The text below the cursor shows the corresponding parameter group, index and bit.	<pre> LOC ↵ PAR EDIT 1002 Ext1 start in1 DI6 [P.02.01.05] CANCEL 00:00 SEL </pre>
7.	<p>Press  to accept any of the preselected values and to return to the parameters list.</p> <p>To freely define a bit of a binary parameter as the value, choose Pointer and press . The parameter group, index and bit will be shown.</p> <p>Select the parameter group with  and . The text below the cursor displays the currently-selected parameter group.</p>	<pre> LOC ↵ PARAMETERS 1002 Ext1 start in1 DI6 1003 Ext1 start in2 1004 Ext2 start func 1005 Ext2 start in1 EXIT 00:00 EDIT </pre> <pre> LOC ↵ PAR EDIT 1002 Ext1 start in1 P.02.01.00 02 I/O values CANCEL 00:00 SAVE </pre>
8.	Press  to select the parameter index. Again, the text below the cursor reflects the current setting.	<pre> LOC ↵ PAR EDIT 1002 Ext1 start in1 P.02.01.00 0201 DI status CANCEL 00:00 SAVE </pre>
9.	Press  to select the bit. Again, the text below the cursor reflects the current setting.	<pre> LOC ↵ PAR EDIT 1002 Ext1 start in1 P.02.01.01 01 DI2 CANCEL 00:00 SAVE </pre>
10.	To save the new value for the pointer parameter, press  . The new value is shown in the parameters list.	<pre> LOC ↵ PARAMETERS 1002 Ext1 start in1 P.02.01.01 1003 Ext1 start in2 1004 Ext2 start func 1005 Ext2 start in1 EXIT 00:00 EDIT </pre>

How to change the value of bit pointer parameter to fixed 0 (FALSE) or 1 (TRUE)

The bit pointer parameter can be fixed to constant value of 0 (FALSE) or 1 (TRUE).

When adjusting a bit pointer parameter on the control panel, CONST is selected in order to fix the value to 0 (displayed as C.FALSE) or 1 (C.TRUE).

Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	<pre> LOC ↺ MAIN MENU — 1 PARAMETERS ASSISTANTS CHANGED PAR EXIT 00:00 ENTER </pre>
2.	Go to the Parameters option by selecting PARAMETERS on the menu with keys  and  , and pressing  . Select the appropriate parameter group with keys  and  . Here the bit pointer parameter <i>14.07 DIO2 out src</i> is used as an example.	<pre> LOC ↺ PAR GROUPS — 01 01 Actual values 02 I/O values 03 Control values 04 Appl values 06 Drive status EXIT 00:00 SEL LOC ↺ PAR GROUPS — 14 10 Start/stop/dir 11 Start/stop mode 12 Operating mode 13 Analogue inputs 14 Digital I/O EXIT 00:00 SEL </pre>
3.	Press  to select the appropriate parameter group. Select the appropriate parameter with keys  and  . Current value of each parameter is shown below its name.	<pre> LOC ↺ PARAMETERS — 1404 DIO1 Ton 1405 DIO1 Toff 1406 DIO2 conf 1407 DIO2 out src P.06.02.03 EXIT 00:00 EDIT </pre>
4.	Press  . Select CONST with keys  and  .	<pre> LOC ↺ PAR EDIT — 1407 DIO2 out src Pointer CANCEL 00:00 NEXT LOC ↺ PAR EDIT — 1407 DIO2 out src Const CANCEL 00:00 NEXT </pre>

Step	Action	Display
5.	Press  .	<pre> LOC ↻ PAR EDIT 1407 DIO2 out src C.FALSE [0] CANCEL 00:00 SAVE </pre>
6.	Specify a new constant value (TRUE or FALSE) for the bit pointer parameter with keys  and  .	<pre> LOC ↻ PAR EDIT 1407 DIO2 out src C.TRUE [1] CANCEL 00:00 SAVE </pre>
7.	To continue, press  . To cancel the new value and keep the original, press  . The new value is shown in the parameters list.	<pre> LOC ↻ PARAMETERS 1407 DIO2 out src C.TRUE 1408 DIO2 Ton 1409 DIO2 Toff 1410 DIO3 conf EXIT 00:00 EDIT </pre>

How to select the monitored signals

Step	Action	Display
1.	<p>You can select which signals are monitored in the Output mode and how they are displayed with group 56 Panel display parameters. See page 25 for detailed instructions on changing parameter values.</p> <p>Note: If you set one of the parameters <i>56.01...56.03</i> to zero, in the output mode you can see names for the two remaining signals. The names are also shown if you set one of the mode parameters <i>56.04...56.06</i> to <i>Disabled</i>.</p>	<pre> LOC ↻ PAR EDIT 5601 signal1 param 01.03 CANCEL 00:00 NEXT </pre> <pre> LOC ↻ PAR EDIT 5602 signal2 param 01.04 CANCEL 00:00 NEXT </pre> <pre> LOC ↻ PAR EDIT 5603 signal3 param 01.06 CANCEL 00:00 NEXT </pre>

■ Assistants













Assistants are routines that guide you through the essential parameter settings related to a specific task, for example application macro selection, entering the motor data, or reference selection.

In the Assistants mode, you can:

- use assistants to guide you through the specification of a set of basic parameters
- start, stop, change the direction and switch between local and remote control.

How to use an assistant

The table below shows how assistants are invoked. The Motor Set-up Assistant is used here as an example.









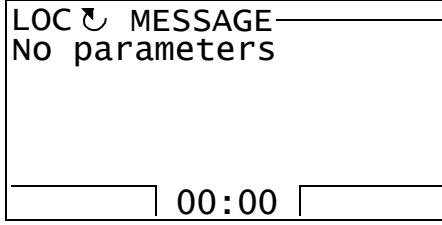
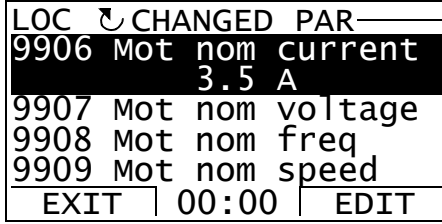

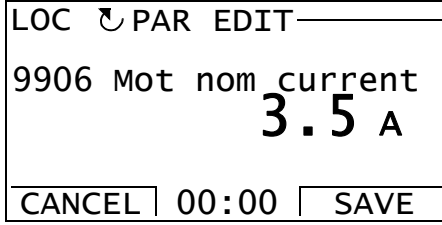


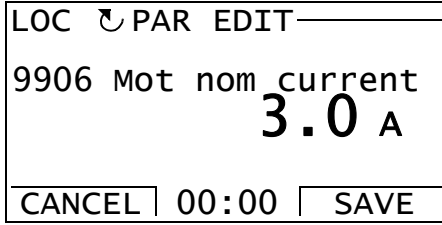
Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	<pre> LOC ↶ MAIN MENU ——— 1 PARAMETERS ASSISTANTS CHANGED PAR EXIT 00:00 ENTER </pre>
2.	Go to the Assistants mode by selecting ASSISTANTS on the menu with keys  and  , and pressing  .	<pre> LOC ↶ CHOICE ——— 1/5 Select assistant Application Macro Motor Set-up Start/Stop Control Reference select EXIT 00:00 OK </pre>
3.	The Motor Set-up assistant is used as an example. Select Motor Set-up with keys  and  , and press  .	<pre> LOC ↶ PAR EDIT ——— 9904 Motor type AM [0] EXIT 00:00 SAVE </pre>
4.	Select the appropriate motor type with keys  and  .	<pre> LOC ↶ PAR EDIT ——— 9904 Motor type PMSM [1] EXIT 00:00 SAVE </pre>
5.	To accept the new value and continue to the setting of the next parameter, press  .	<pre> LOC ↶ PAR EDIT ——— 9905 Motor ctrl mode DTC [0] EXIT 00:00 SAVE </pre>
	After all the parameters of the assistant are set, the main menu is displayed. To run another assistant, repeat the procedure from step 2. To abort an assistant, press  at any point.	

■ Changed Parameters



In the Changed Parameters mode, you can:

- view a list of all parameters that have been changed from the macro default values
- change these parameters
- start, stop, change the direction and switch between local and remote control.

How to view and edit changed parameters

Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	
2.	Go to the Changed Parameters mode by selecting CHANGED PAR on the menu with keys  and  , and pressing  . If there are no changed parameters in the history, corresponding text will be shown. If parameters have been changed, a list of them is shown. Select the changed parameter on the list with keys  and  . The value of the selected parameter is shown below it.	 
3.	Press  to modify the value.	
4.	Specify a new value for the parameter with keys  and  . Pressing the key once increments or decrements the value. Holding the key down changes the value faster. Pressing the keys simultaneously replaces the displayed value with the default value.	

34 The ACS850 control panel








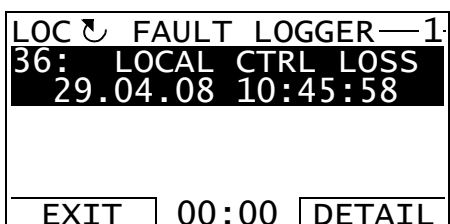






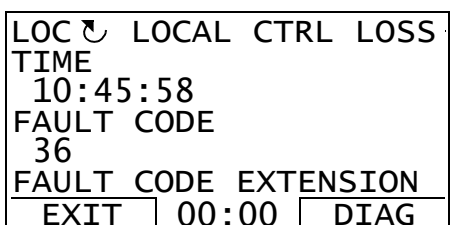

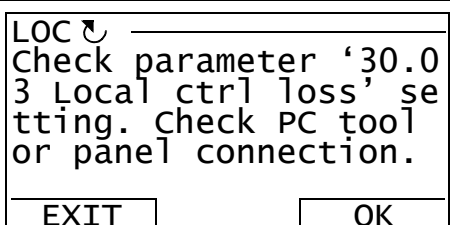
Step	Action	Display
5.	<p>To accept the new value, press . If the new value is the default value, the parameter is removed from the list of changed parameters.</p> <p>To cancel the new value and keep the original, press .</p>	<pre> LOC ↶ CHANGED PAR 9906 Mot nom current 3.0 A 9907 Mot nom voltage 9908 Mot nom freq 9909 Mot nom speed EXIT 00:00 EDIT </pre>


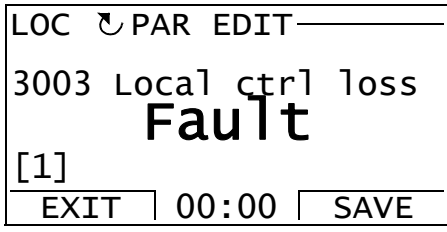




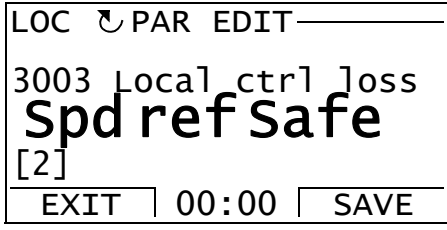
■ Fault Logger

In the Fault Logger option, you can:



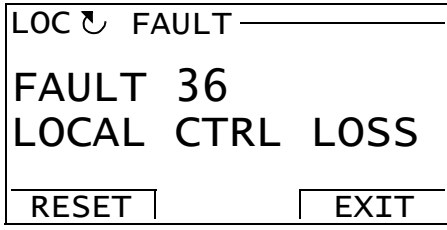
- view the drive fault history
- see the details of the most recent faults
- read the help text for the fault and make corrective actions
- start, stop, change the direction and switch between local and remote control.

How to view faults

Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	
2.	Go to the Fault Logger option by selecting FAULT LOGGER on the menu with keys  and  , and pressing  . If there are no faults in the fault history, corresponding text will be shown. If there is a fault history, the display shows the fault log starting with the most recent fault. The number on the row is the fault code according to which the causes and corrective actions are listed in chapter Fault tracing (page 283).	 
3.	To see the details of a fault, select it with keys  and  , and press  . Scroll the text with keys  and  . To return to the previous display, press  .	
4.	If you want help in diagnosing the fault, press  .	

Step	Action	Display
5.	Press  . The panel allows you to edit necessary parameters to correct the fault.	
6.	Specify a new value for the parameter with keys  and  . To accept the new value, press  . To cancel the new value and keep the original, press  .	

How to reset faults

Step	Action	Display
1.	When a fault occurs, a text identifying the fault is shown. To reset the fault, press  . To return to the previous display, press  .	







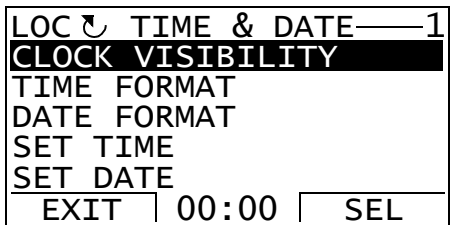





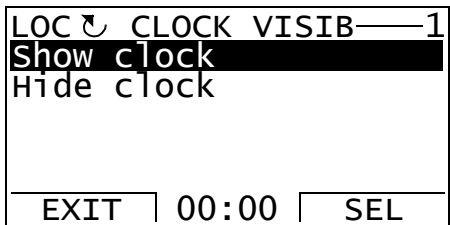





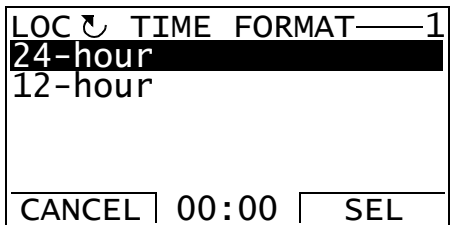
■ Time & Date





























In the Time & Date option, you can:

- show or hide the clock
- change date and time display formats
- set the date and time
- enable or disable automatic clock transitions according to the daylight saving changes
- start, stop, change the direction and switch between local and remote control.

The Control Panel contains a battery to ensure the function of the clock when the panel is not powered by the drive.

How to show or hide the clock, change display formats, set the date and time and enable or disable clock transitions due to daylight saving changes

Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	
2.	Go to the Time & Date option by selecting TIME & DATE on the menu with keys  and  , and pressing  .	
3.	To show (hide) the clock, select CLOCK VISIBILITY on the menu, press  , select Show clock (Hide clock) with keys  and  and press  , or, if you want to return to the previous display without making changes, press  .	
	To specify the time format, select TIME FORMAT on the menu, press  , and select a suitable format with keys  and  . Press  to save or  to cancel your changes.	

Step	Action	Display
	<p>To specify the date format, select DATE FORMAT on the menu, press  and select a suitable format. Press  to save or  to cancel your changes.</p> <p>To set the time, select SET TIME on the menu and press . Specify the hours with keys  and , and press . Then specify the minutes. Press  to save or  to cancel your changes.</p>	<p>LOC  DATE FORMAT—1 dd.mm.yy mm/dd/yy dd.mm.yyyy mm/dd/yyyy</p> <p>CANCEL 00:00 OK</p> <hr/> <p>LOC  SET TIME— 15:41</p> <p>CANCEL OK</p>
	<p>To set the date, select SET DATE on the menu and press . Specify the first part of the date (day or month depending on the selected date format) with keys  and , and press . Repeat for the second part. After specifying the year, press . To cancel your changes, press .</p>	<p>LOC  SET DATE— 19.03.2008</p> <p>CANCEL 00:00 OK</p>
	<p>To enable or disable the automatic clock transitions according to the daylight saving changes, select DAYLIGHT SAVING on the menu and press . Pressing  opens the help that shows the beginning and end dates of the period during which daylight saving time is used in each country or area whose daylight saving changes you can select to be followed. Scroll the text with keys  and . To return to the previous display, press .</p> <p>To disable automatic clock transitions according to the daylight saving changes, select Off and press .</p> <p>To enable automatic clock transitions, select the country or area whose daylight saving changes are followed and press .</p> <p>To return to the previous display without making changes, press .</p>	<p>LOC  DAYLIGHT SAV—1 Off EU US Australia1:NSW,Vict.. Australia2:Tasmania..</p> <p>EXIT 00:00 SEL</p> <hr/> <p>LOC  HELP— EU: On: Mar last Sunday Off: Oct last Sunday</p> <p>US: EXIT 00:00</p>

■ Parameter Backup

The Parameter Backup option is used to export parameters from one drive to another or to make a backup of the drive parameters. Uploading stores all drive parameters, including up to four user sets, to the Control Panel. Selectable subsets of the backup file can then be restored/downloaded from the control panel to the same drive or another drive of the same type.

In the Parameter Backup option, you can:



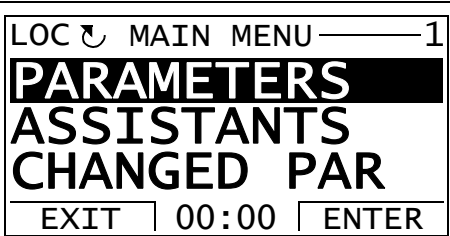
- Copy all parameters from the drive to the control panel with MAKE BACKUP TO PANEL. This includes all defined user sets of parameters and internal (not adjustable by the user) parameters such as those created by the ID Run.
- View the information about the backup stored in the control panel with SHOW BACKUP INFO. This includes e.g. version information etc. of the current backup file in the panel. It is useful to check this information when you are going to restore the parameters to another drive with RESTORE PARS ALL to ensure that the drives are compatible.
- Restore the full parameter set from the control panel to the drive using the RESTORE PARS ALL command. This writes all parameters, including the internal non-user-adjustable motor parameters, to the drive. It does NOT include the user sets of parameters.
















Note: Use this function only to restore the parameters from a backup or to restore parameters to systems that are compatible.

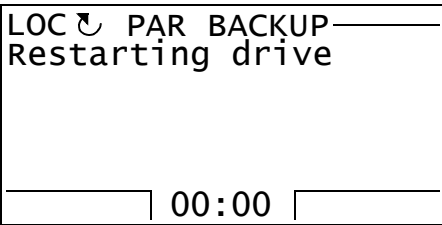
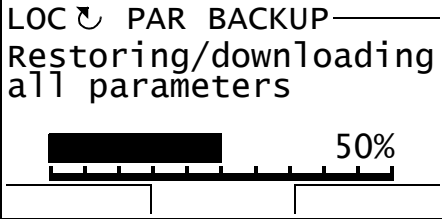
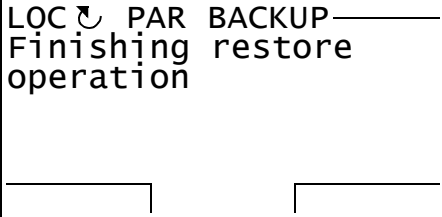
- Restore all parameters, except motor data, to the drive with RESTORE PARS NO-IDRUN.
- Restore only motor data parameters to the drive with RESTORE PARS IDRUN.
- Restore all user sets to the drive with RESTORE ALL USER SETS.
- Restore only user set 1...4 to the drive with RESTORE USER SET 1...RESTORE USER SET 4.

How to backup and restore parameters

For all backup and restore functions available, see page [39](#).

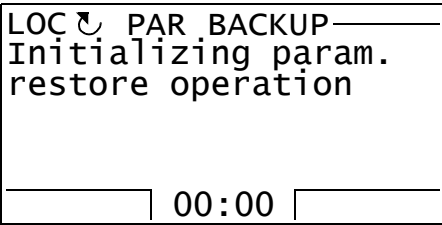




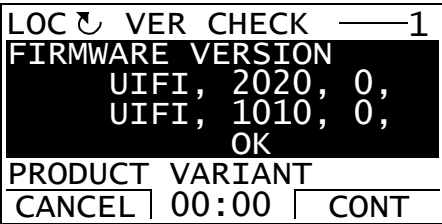
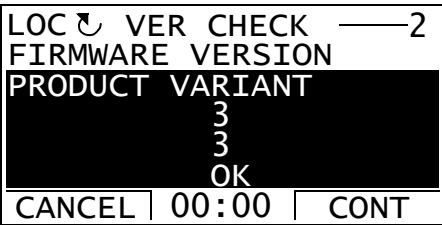
Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	

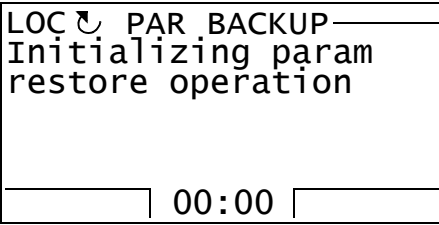
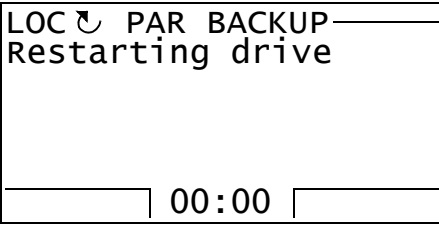
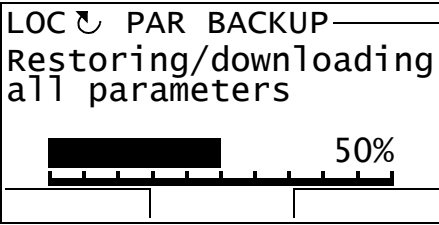
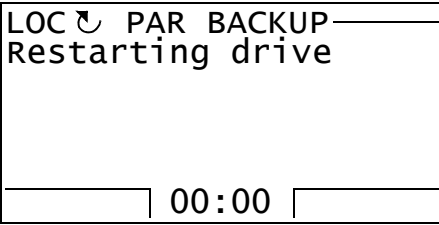
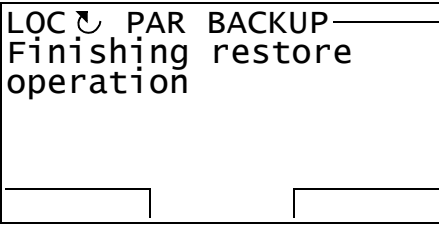


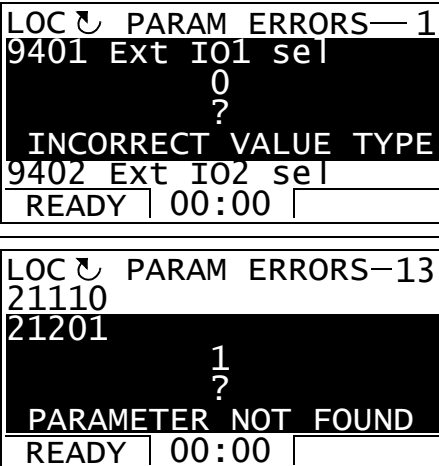
Step	Action	Display
2.	<p>Go to the Parameter Backup option by selecting PAR BACKUP on the menu with keys  and , and pressing .</p>	<pre> LOC ↺ PAR BACKUP——1 MAKE BACKUP TO PANEL SHOW BACKUP INFO RESTORE PARS ALL RESTORE PARS NO-IDRUN RESTORE PARS IDRUN EXIT 00:00 SEL </pre>
	<p>To copy all parameters (including user sets and internal parameters) from the drive to the control panel, select MAKE BACKUP TO PANEL on the Par Backup with keys  and , and press . Operation starts. Press  if you want to stop the operation.</p> <p>After the backup is completed, the display shows a message about the completion. Press  to return to the Par Backup.</p>	<pre> LOC ↺ PAR BACKUP—— Copying file 1/2 ABORT 00:00 LOC ↺ MESSAGE—— Parameter upload successful OK 00:00 </pre>
	<p>To perform restore functions, select the appropriate operation (here RESTORE PARS ALL is used as an example) on the Par Backup with keys  and .</p> <p>Press . Restoring starts.</p> <p>Backup interface version is checked. Scroll the text with keys  and .</p> <p>If you want to continue, press . Press  if you want to stop the operation. If the downloading is continued, the display shows a message about it.</p>	<pre> LOC ↺ PAR BACKUP——3 MAKE BACKUP TO PANEL SHOW BACKUP INFO RESTORE PARS ALL RESTORE PARS NO-IDRUN RESTORE PARS IDRUN EXIT 00:00 SEL LOC ↺ PAR BACKUP—— Initializing param restore operation 00:00 LOC ↺ VERSION CHECK -1 BACKUP INTERFACE VER 0.2 0.2 OK FIRMWARE VERSION CANCEL 00:00 CONT LOC ↺ PAR BACKUP—— Initializing param. restore operation 00:00 </pre>





Step	Action	Display
	Downloading continues, drive is being restarted.	
	<p>The display shows the transfer status as a percentage of completion.</p> <p>Downloading finishes.</p>	 

Parameter errors

If you try to backup and restore parameters between different firmware versions, the panel shows you the following parameter error information:



Step	Action	Display
1.	Restore operation starts normally.	
2.	<p>Firmware version is checked. You can see on the panel that the firmware versions are not the same.</p> <p>Scroll the text with keys  and . To continue, press . Press  to stop the operation.</p>	 

Step	Action	Display
3.	If the downloading is continued, the display shows a message about it.	
	Downloading continues, drive is being restarted.	
	The display shows the transfer status as a percentage of completion.	
	Downloading continues.	
	Downloading finishes.	
4.	<p>The panel shows a list of erroneous parameters.</p> <p>You can scroll the parameters with keys  and . The reason for parameter error is also shown.</p>	

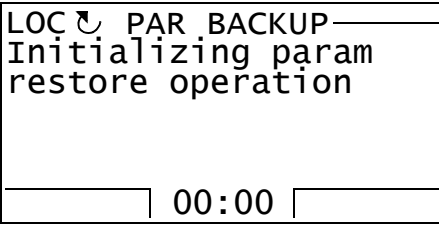
Step	Action	Display
5.	You can edit parameters by pressing  when EDIT command is visible. Parameter <i>95.01 Ctrl boardSupply</i> is used as an example. Edit the parameter as shown in section <i>Parameters</i> on page 25.	<pre> LOC ↻ PAR EDIT 9501 Ctrl boardSupply External 24V [1] CANCEL 00:00 SAVE </pre>
6.	Press  to save the new value. Press  to return to the list of erroneous parameters.	<pre> LOC ↻ PAR EDIT 9501 Ctrl boardSupply Internal 24V [0] CANCEL 00:00 SAVE </pre>
7.	The parameter value you chose is visible under the parameter name. Press  when you have finished editing the parameters.	<pre> LOC ↻ PARAM ERRORS—9 9501 Ctrl boardSupply 0 0 INCORRECT VALUE TYPE 9503 READY 00:00 EDIT </pre>

Trying to restore a user set between different firmware versions

If you try to backup and restore a user set between different firmware versions, the panel shows you the following alarm information:










Step	Action	Display
1.	Restore operation starts normally.	<pre> LOC ↻ PAR BACKUP— Initializing param restore operation 00:00 </pre>
2.	Version check is also OK. You can see on the panel that the firmware versions are not the same. You can scroll the text with keys  and  .	<pre> LOC ↻ VER CHECK —1 FIRMWARE VERSION UIFI, 2020, 0, UIFI, 1010, 0, OK PRODUCT VARIANT CANCEL 00:00 CONT </pre> <pre> LOC ↻ VER CHECK —2 FIRMWARE VERSION PRODUCT VARIANT 3 3 OK CANCEL 00:00 CONT </pre>

44 The ACS850 control panel



Step	Action	Display
3.	If the downloading is continued, the display shows a message about it.	<div style="border: 1px solid black; padding: 5px;"> <p>LOC ↻ PAR BACKUP</p> <p>Initializing param restore operation</p> <hr/> <p style="text-align: center;">00:00</p> </div>
4.	Downloading continues, drive is being restarted.	<div style="border: 1px solid black; padding: 5px;"> <p>LOC ↻ PAR BACKUP</p> <p>Restarting drive</p> <hr/> <p style="text-align: center;">00:00</p> </div>
5.	The display shows the transfer status as a percentage of completion.	<div style="border: 1px solid black; padding: 5px;"> <p>LOC ↻ PAR BACKUP</p> <p>Restoring/downloading user set 1</p> <div style="text-align: center; margin-top: 10px;">  <p>50%</p> </div> <hr/> </div>
6.	Downloading continues.	<div style="border: 1px solid black; padding: 5px;"> <p>LOC ↻ PAR BACKUP</p> <p>Initializing param restore operation</p> <hr/> <p style="text-align: center;">00:00</p> </div>
7.	Downloading continues, drive is being restarted.	<div style="border: 1px solid black; padding: 5px;"> <p>LOC ↻ PAR BACKUP</p> <p>Restarting drive</p> <hr/> <p style="text-align: center;">00:00</p> </div>
8.	Downloading finishes.	<div style="border: 1px solid black; padding: 5px;"> <p>LOC ↻ PAR BACKUP</p> <p>Finishing restore operation</p> <hr/> </div>
9.	Panel shows a text identifying the alarm and returns to the Par Backup.	<div style="border: 1px solid black; padding: 5px;"> <p>LOC ↻ ALARM</p> <p>ALARM 2036 RESTORE</p> <hr/> <p style="text-align: center;">EXIT</p> </div>










Trying to load a user set between different firmware versions

If you try load a user set between different firmware versions, the panel shows you the following fault information:

Step	Action	Display
1.	Go to the Parameters option by selecting PARAMETERS on the main menu as shown in section Parameters on page 25. A user set is loaded through parameter 16.09 User set sel. Select parameter group 16 System with keys  and  .	<pre> LOC ↻ PAR GROUPS——16 12 Operating mode 13 Analogue inputs 14 Digital I/O 15 Analogue outputs 16 System EXIT 00:00 SEL </pre>
2.	Press  to select parameter group 16. Select parameter 16.09 User set sel with keys  and  . Current value of each parameter is shown below its name.	<pre> LOC ↻ PARAMETERS—— 1603 Pass code 1604 Param restore 1607 Param save 1609 User set sel No request EXIT 00:00 EDIT </pre>
3.	Press  . Select the user set you want to load with keys  and  . Press  .	<pre> LOC ↻ PAR EDIT—— 1609 User set sel No request [1] CANCEL 00:00 SAVE </pre> <pre> LOC ↻ PAR EDIT—— 1609 User set sel Load set 1 [2] CANCEL 00:00 SAVE </pre>
4.	Panel shows a text identifying the fault.	<pre> LOC ↻ FAULT—— FAULT 310 USERSET LOAD RESET EXIT </pre>

How to view information about the backup

Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	<pre> LOC ↻ MAIN MENU——1 PARAMETERS ASSISTANTS CHANGED PAR EXIT 00:00 ENTER </pre>















Step	Action	Display
2.	Go to the Par Backup option by selecting PAR BACKUP on the menu with keys  and  , and pressing  . Select SHOW BACKUP INFO with keys  and  .	<pre> LOC ↺ PAR BACKUP——2 MAKE BACKUP TO PANEL SHOW BACKUP INFO RESTORE PARS ALL RESTORE PARS NO-IDRUN RESTORE PARS IDRUN EXIT 00:00 SEL </pre>
3.	Press  . The display shows the following information about the drive from where the backup was made: BACKUP INTERFACE VER: Format version of the backup file FIRMWARE VERSION: Information on the firmware UIFI: Firmware of the ACS850 drive 2020: Firmware version 0: Firmware patch version PRODUCT VARIANT: 3: ACS850 (Standard control program) You can scroll the information with  and  .	<pre> LOC ↺ BACKUP INFO—— BACKUP INTERFACE VER 0.4 0.4 FIRMWARE VERSION UIFI,2020,0, EXIT 00:00 </pre> <pre> LOC ↺ BACKUP INFO—— FIRMWARE VERSION UIFI,2020,0, UIFI,1010,0, PRODUCT VARIANT 3 EXIT 00:00 </pre>
4.	Press  to return to the Par Backup.	<pre> LOC ↺ PAR BACKUP——1 MAKE BACKUP TO PANEL SHOW BACKUP INFO RESTORE PARS ALL RESTORE PARS NO-IDRUN RESTORE PARS IDRUN EXIT 00:00 SEL </pre>








■ I/O Settings

In the I/O Settings mode, you can:

- check the parameter settings that configure the I/Os of the drive
- check the parameters that have an input or output selected as their source or target
- edit the parameter setting
- start, stop, change the direction and switch between local and remote control.

How to edit and change parameter settings related to I/O terminals

Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	<pre> LOC ↻ MAIN MENU — 1 PARAMETERS ASSISTANTS CHANGED PAR EXIT 00:00 ENTER </pre>
2.	Go the I/O Settings mode by selecting I/O SETTINGS on the menu with keys  and  , and pressing  . Select the I/O group, e.g. Digital inputs, with keys  and  .	<pre> LOC ↻ I/O SETTINGS — 1 Analog outputs Analog inputs Digital I/Os Digital inputs Relay outputs EXIT 00:00 SEL LOC ↻ I/O SETTINGS — 4 Analog outputs Analog inputs Digital I/Os Digital inputs Relay outputs EXIT 00:00 SEL </pre>
3.	Press  . After a brief pause, the display shows the current settings for the selection. You can scroll digital inputs and parameters with keys  and  .	<pre> LOC ↻ I/O SETTINGS — 1 DI1 1002 Ext1 start in1 DI2 DI3 1010 Fault reset sel EXIT 00:00 INFO </pre>
4.	Press  . The panel shows information related to I/O selected (in this case, DI1). You can scroll information with keys  and  . Press  to return to the digital inputs.	<pre> LOC ↻ I/O INFO — NUM OF I/O ITEMS 0 SLOT NUMBER 0 NODE NUMBER EXIT 00:00 </pre>

Step	Action	Display
5.	Select the setting (line with a parameter number) with keys  and  . You can edit the parameter (INFO selection turns into EDIT selection).	<pre> LOC ↶ I/O SETTINGS—1 DI1 1002 Ext1 start in1 DI2 DI3 1010 Fault reset sel ----- EXIT 00:00 EDIT </pre>
6.	Press  .	<pre> LOC ↶ PAR EDIT— 1002 Ext1 start in1 DI1 [P.02.01.00] ----- CANCEL 00:00 SEL </pre>
7.	Specify a new value for the setting with keys  and  . Pressing the key once increments or decrements the value. Holding the key down changes the value faster. Pressing the keys simultaneously replaces the displayed value with the default value.	<pre> LOC ↶ PAR EDIT— 1002 Ext1 start in1 DI04 [P.02.03.03] ----- CANCEL 00:00 SEL </pre>
8.	To save the new value, press  . To cancel the new value and keep the original, press  .	<pre> LOC ↶ I/O SETTINGS—1 DI1 1002 Ext1 start in1 DI2 DI3 1010 Fault reset sel ----- EXIT 00:00 EDIT </pre>


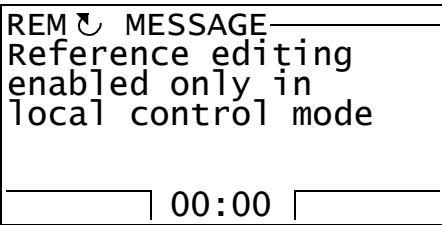






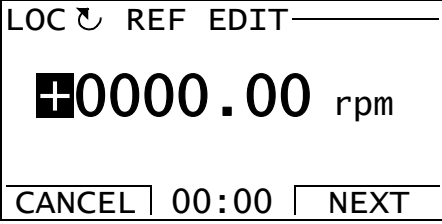






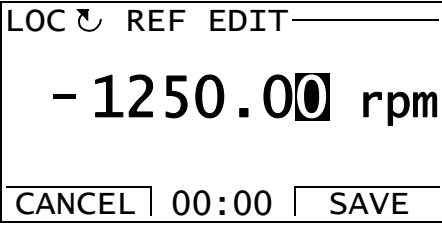


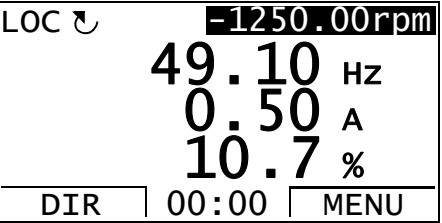
■ Reference Edit

In the Reference Edit option, you can:

- accurately control the local reference value,
- start, stop, change the direction and switch between local and remote control.

Editing is allowed only in the LOC state, the option always edits the local reference value.

How to edit reference value









Step	Action	Display
1.	If the panel is in the remote control mode (REM shown on the status line), switch to local control (LOC shown on the status line) by pressing  . Reference editing is not possible in remote control mode. (See page 22 for more information on switching between the local and remote control modes.) The display shows a message about that if you try to enter REF EDIT in the remote control mode.	
2.	Otherwise, go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	
3.	Go to the Reference Edit option by selecting REF EDIT on the menu with keys  and  , and pressing  .	
4.	Select the correct sign with keys  and  , and press  . Select the correct numbers with keys  and  , and after each number is selected, press  .	
5.	After the last number is selected, press  . Go to the Output mode by pressing  . The selected reference value is shown in the status line.	

■ Drive Info

In the Drive Info option, you can:

- view information on the drive,
- start, stop, change the direction and switch between local and remote control.

How to view drive info









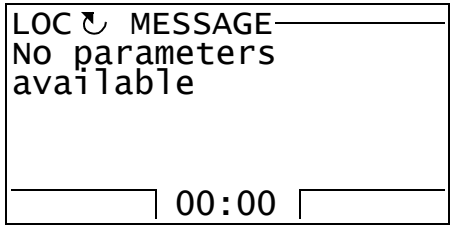
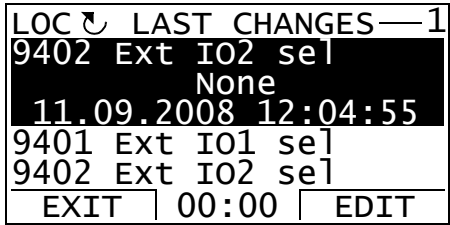



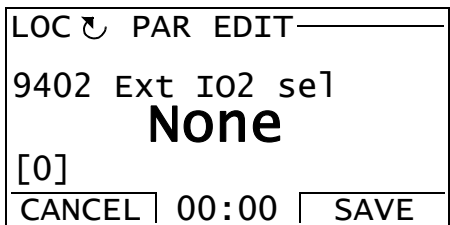




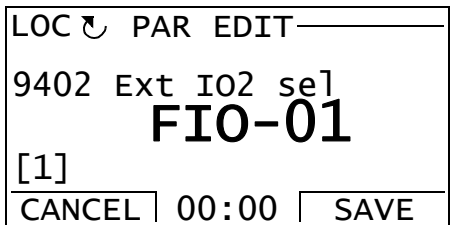
Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	<pre> LOC ↻ MAIN MENU ——— 1 PARAMETERS ASSISTANTS CHANGED PAR EXIT 00:00 ENTER </pre>
2.	Go to the Drive info option by selecting DRIVE INFO on the menu with keys  and  , and pressing  .	<pre> LOC ↻ DRIVE INFO ——— DRIVE NAME - DRIVE TYPE ACS850 DRIVE MODEL EXIT 00:00 </pre>
3.	The display shows information about the drive. You can scroll the information with keys  and  . Note: The information shown may vary according to the firmware version of the drive. DRIVE NAME: Drive name defined as a text in DriveStudio commissioning and maintenance tool DRIVE TYPE: e.g. ACS850 DRIVE MODEL: Type code of the drive FW VERSION: See page 45 . SOLUTION PROGRAM: Version information of the active application program BASE SOLUTION PROGRAM: Version information of the application program template STANDARD LIBRARY: Version information of the standard library TECHNOLOGY LIBRARY: Not applicable to the ACS850 POWER UNIT SERNO: Serial number of the power stage (JPU) MEM UNIT HW SERNO: Serial number in manufacturing the memory unit (JMU) MEM UNIT CONFIG SERNO: Serial number in configuring the memory unit (JMU). Press  to return to the Main menu.	<pre> LOC ↻ DRIVE INFO ——— FW VERSION UIFI, 2020, 0, SOLUTION PROGRAM - BASE SOLUTION PROGRAM EXIT 00:00 </pre>

■ Parameter Change Log

In the Parameter Change Log option, you can:

- view latest parameter changes made via control panel or PC tool,
- edit these parameters,
- start, stop, change the direction and switch between local and remote control.

How to view latest parameter changes and edit parameters

Step	Action	Display
1.	Go to the Main menu by pressing  if you are in the Output mode. Otherwise press  repeatedly until you get to the Main menu.	
2.	Go to the Parameter Change Log option by selecting PAR CHG LOG on the menu with keys  and  , and pressing  . If there are no parameter changes in the history, corresponding text will be shown. If there are parameter changes in the history, the panel shows a list of the last parameter changes starting from the most recent change. The order of the changes is also indicated with a number in the top right corner (1 stands for most recent change, 2 the second latest change etc.) If a parameter has been changed twice, it is shown as one change in the list. The current value of the parameter and the parameter change date and time are also shown below the selected parameter. You can scroll the parameters with keys  and  .	 
3.	If you want to edit a parameter, select the parameter with keys  and  and press  .	
4.	Specify a new value for the parameter with keys  and  . To save the new value, press  . To cancel the new value and keep the original, press  .	

52 The ACS850 control panel

Step	Action	Display
5.	<p>The parameter change is shown as the first one in the list of last parameter changes.</p> <p>Note: You can reset the parameter change log by setting parameter 16.14 Reset ChgParLog to Reset.</p>	<pre> LOC ↻ LAST CHANGES—1 9402 Ext IO2 sel FIO-01 12.09.2008 15:09:33 9402 Ext IO2 sel 9401 Ext IO1 sel EXIT 00:00 EDIT </pre>



Control locations and operating modes

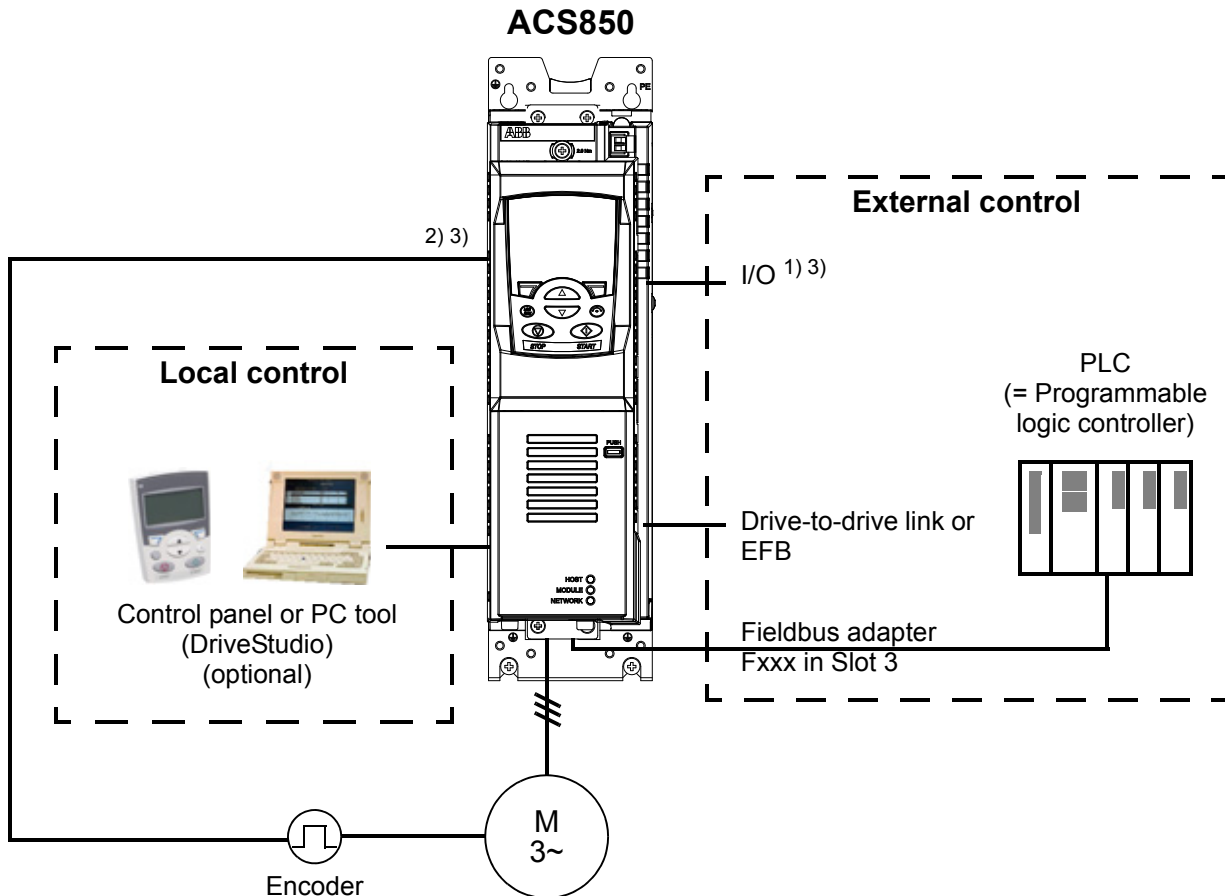
What this chapter contains

This chapter describes the control locations and operating modes of the drive.

efesotomasyon.com

Local control vs. external control

The drive has two main control locations: external and local. The control location is selected with the LOC/REM key on the control panel or with the PC tool (Take/Release button).



1) Extra inputs/outputs can be added by installing optional I/O extension modules (FIO-xx) in drive Slot 1/2.

2) Encoder or resolver interface module (FEN-xx) installed in drive Slot 1/2

3) Two encoder/resolver interface modules of the same type are not allowed.

■ Local control

The control commands are given from the control panel keypad or from a PC equipped with DriveStudio when the drive is in local control. Speed and torque control modes are available for local control.

Local control is mainly used during commissioning and maintenance. The control panel always overrides the external control signal sources when used in local control. Changing the control location to local can be disabled by parameter [16.01 Local lock](#).

The user can select by a parameter ([30.03 Local ctrl loss](#)) how the drive reacts to a control panel or PC tool communication break.

■ External control

When the drive is in external control, control commands are given through the fieldbus interface (via an embedded fieldbus interface or an optional fieldbus adapter module), the I/O terminals (digital and analogue inputs), optional I/O extension modules or the drive-to-drive link. External references are given through the fieldbus interface, analogue inputs, drive to drive link and encoder inputs.

Two external control locations, EXT1 and EXT2, are available. The user can select control signals (e.g. start and stop) and control modes for both external control locations. Depending on the user selection, either EXT1 or EXT2 is active at a time. Selection between EXT1/EXT2 is done via digital inputs or fieldbus control word.

Operating modes of the drive

The drive can operate in several control modes.

■ Speed control mode

Motor rotates at a speed proportional to the speed reference given to the drive. This mode can be used either with estimated speed used as feedback, or with an encoder or resolver for better speed accuracy.

Speed control mode is available in both local and external control.

■ Torque control mode

Motor torque is proportional to the torque reference given to the drive. This mode can be used either with or without an encoder or resolver. When used with an encoder or resolver, this mode provides for more accurate and dynamic motor control.

Torque control mode is available in both local and external control.

■ Special control modes

In addition to the above-mentioned control modes, the following special control modes are available:

- Emergency stop modes OFF1 and OFF3: Drive stops along the defined deceleration ramp and drive modulation stops.
- Jogging mode: Drive starts and accelerates to the defined speed when the jogging signal is activated.

For more information, see parameter group [10 Start/stop/dir](#) on page [122](#).

4

Program features

efesotomasyon.com

What this chapter contains

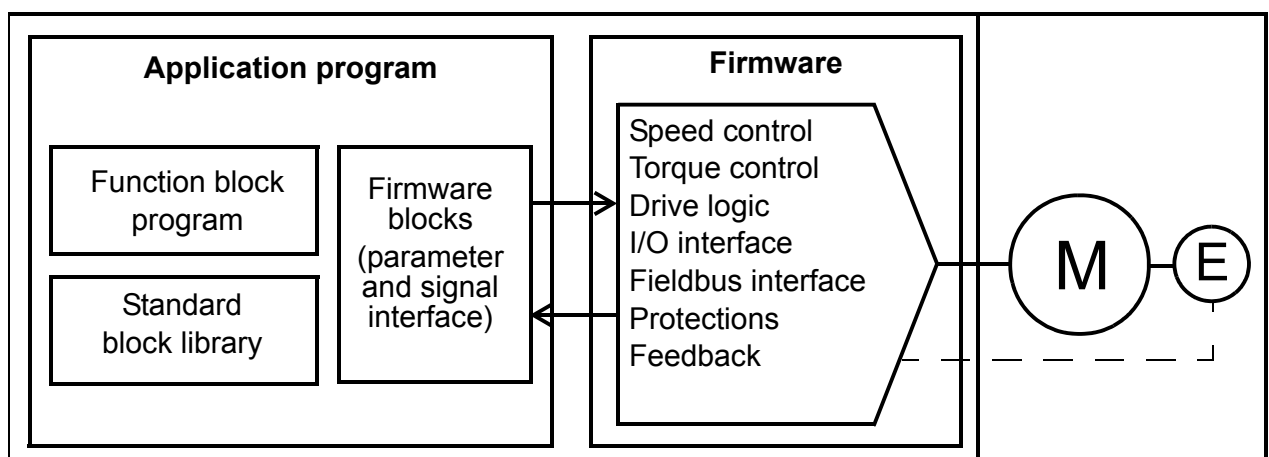
This chapter describes the features of the control program.

Drive configuration and programming

The drive control program is divided into two parts:

- firmware program
- application program.

Drive control program



The firmware program performs the main control functions, including speed and torque control, drive logic (start/stop), I/O, feedback, communication and protection functions. Firmware functions are configured and programmed with parameters.

■ Programming via parameters

Parameters can be set via

- the control panel, as described in chapter [The ACS850 control panel](#)
- the DriveStudio PC tool, as described in *DriveStudio User Manual* (3AFE68749026 [English]), or
- the fieldbus interface, as described in chapters [Control through the embedded fieldbus interface](#) and [Control through a fieldbus adapter](#).

All parameter settings are stored automatically to the permanent memory of the drive. However, if an external +24 V DC power supply is used for the drive control unit, it is highly recommended to force a save by using parameter [16.07 Param save](#) before powering down the control unit after any parameter changes.

If necessary, the default parameter values can be restored by parameter [16.04 Param restore](#).

Note: By default, a selective list of parameters is shown by the drive panel or DriveStudio. All parameters can be displayed by setting parameter [16.15 Menu set sel](#) to [Load long](#).

■ Application programming

The functions of the firmware program can be extended with application programming. (A standard drive delivery does not include an application program.) Application programs can be built out of function blocks based on the IEC-61131 standard. Some drive parameters are used as firmware function block inputs and can therefore be modified also via the application program.

For more information, see

- *Application guide: Application programming for ACS850 drives* (3AUA0000078664 [English]), and
- *DriveSPC User manual* (3AFE68836590 [English]).

Application program licensing and protection

The drive can be assigned an application licence consisting of an ID and password using the DriveSPC tool. Likewise, the application program created in DriveSPC can be protected by an ID and password.

If a protected application program is downloaded to a licensed drive, the IDs and passwords of the application and drive must match. A protected application cannot be downloaded to an unlicensed drive. On the other hand, an unprotected application can be downloaded to a licensed drive.

The ID of the application licence is displayed by DriveStudio in the drive software properties as APPL LICENCE. If the value is 0, no licence has been assigned to the drive.

Notes:

- The application licence can only be assigned to a complete drive, not a stand-alone control unit.
- A protected application can only be downloaded to a complete drive, not a stand-alone control unit.

Control interfaces

■ Programmable analog inputs

The drive has two programmable analog inputs. Each of the inputs can be independently set as a voltage (0/2...10 V or -10...10 V) or current (0/4...20 mA) input by a jumper on the JCU Control Unit. Each input can be filtered, inverted and scaled. The number of analog inputs can be increased by using FIO-xx I/O extensions.

Settings

Parameter group [13 Analogue inputs](#) (page 133).

■ Programmable analog outputs

The drive has two current analog outputs. Each output can be filtered, inverted and scaled. The number of analog outputs can be increased by using FIO-xx I/O extensions.

Settings

Parameter group [15 Analogue outputs](#) (page 153).

■ Programmable digital inputs and outputs

The drive has six digital inputs, a digital start interlock input, and two digital input/outputs.

One digital input (DI6) doubles as a PTC thermistor input. See section [Thermal motor protection](#) on page 80.

One of the digital input/outputs can be used as a frequency input, one as a frequency output.

The number of digital inputs/outputs can be increased by using FIO-xx I/O extensions.

Settings

Parameter group [14 Digital I/O](#) (page 140).

■ Programmable I/O extensions

The number of inputs and outputs can be increased by using FIO-xx I/O extensions. The drive I/O configuration parameters (parameter groups 13, 14 and 15) include the maximum number of DI, DIO, AI, AO and RO that can be taken into use with different FIO-xx combinations.

The table below shows the possible I/O combinations of the drive:

Location	Digital inputs (DI)	Digital I/O (DIO)	Analog inputs (AI)	Analog outputs (AO)	Relay outputs (RO)
JCU Control Unit	7	2	2	2	2
FIO-01	-	4	-	-	2
FIO-11	-	2	3	1	-
FIO-21	1	-	1	-	2

For example, with FIO-01 and FIO-21 connected to the drive, parameters controlling DI1...8, DIO1...6, AI1...3, AO1...2 and RO1...7 are in use.

Settings

Parameter groups [13 Analogue inputs](#) (page 133), [14 Digital I/O](#) (page 140), [15 Analogue outputs](#) (page 153) and [94 Ext IO conf](#) (page 250).

■ Programmable relay outputs

The drive has three relay outputs. The signal to be indicated by the outputs can be selected by parameters.

Relay outputs can be added by using FIO-xx I/O extensions.

Settings

Parameter group [14 Digital I/O](#) (page 140).

■ Fieldbus control

The drive can be connected to several different automation systems through its fieldbus interface. See chapters [Control through the embedded fieldbus interface](#) (page 303) and [Control through a fieldbus adapter](#) (page 331).

Settings

Parameter groups [50 Fieldbus](#) (page 230), [51 FBA settings](#) (page 232), [52 FBA data in](#) (page 233), [53 FBA data out](#) (page 234) and [58 Embedded Modbus](#) (page 237).

Motor control

■ Constant speeds

It is possible to predefine up to 7 constant speeds. Constant speeds can be activated, for example, through digital inputs. Constant speeds override the speed reference.

Settings

Parameter group [26 Constant speeds](#) (page [183](#)).

■ Critical speeds

A Critical speeds function is available for applications where it is necessary to avoid certain motor speeds or speed ranges because of, for example, mechanical resonance problems.

Settings

Parameter group [25 Critical speed](#) (page [182](#)).

■ Speed controller tuning

The speed controller of the drive can be automatically adjusted using the autotune function (parameter [23.20 PI tune mode](#)). Autotuning is based on the load and inertia of the motor and the machine. It is, however, also possible to manually adjust the controller gain, integration time and derivation time.

Autotuning can be performed in four different ways depending on the setting of parameter [23.20 PI tune mode](#). The selections *Smooth*, *Middle* and *Tight* define how the drive torque reference should react to a speed reference step after tuning. The selection *Smooth* will produce a slow response; *Tight* will produce a fast response. The selection *User* allows customized control sensitivity adjustment through parameters [23.21 Tune bandwidth](#) and [23.22 Tune damping](#). Detailed tuning status information is provided by parameter [06.03 Speed ctrl stat](#). If the autotuning routine fails, the *AUTOTUNE FAILED (0x8481)* alarm will occur for approximately 15 seconds. If a stop command is given to the drive during the autotuning, the routine is aborted.

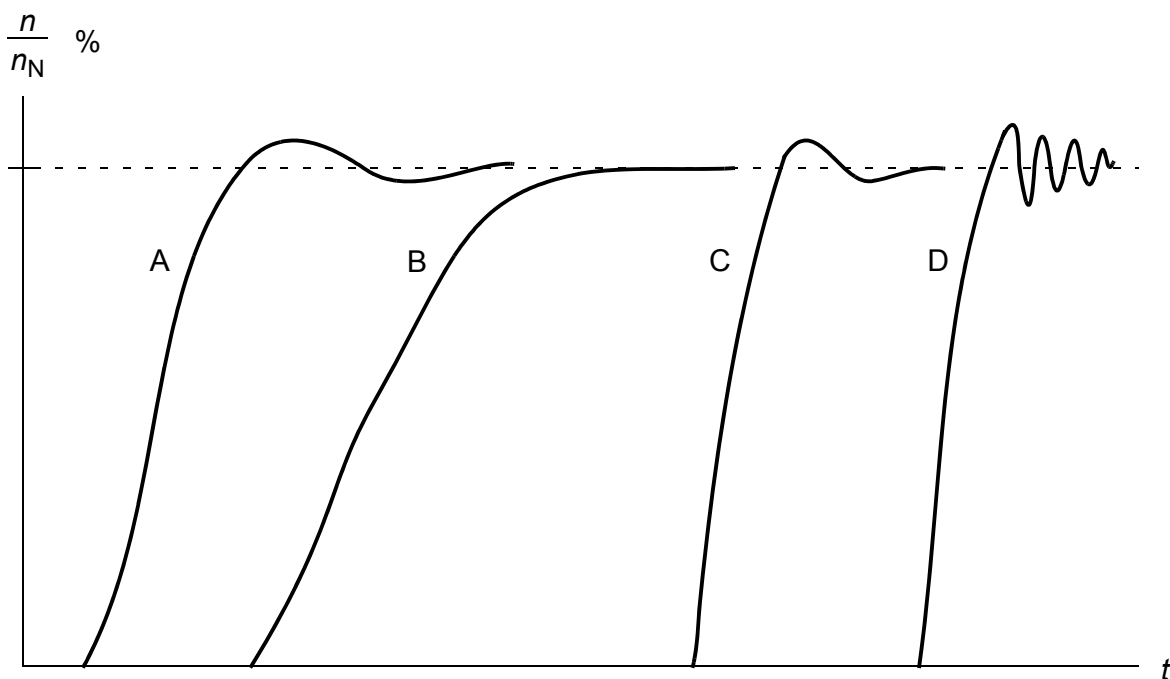
The prerequisites for performing the autotune routine are:

- The ID run has been successfully completed
- Speed, torque, current and acceleration limits (parameter groups [20 Limits](#) and [22 Speed ref ramp](#)) are set
- Speed feedback filtering, speed error filtering and zero speed are set (parameter groups [19 Speed calculation](#) and [23 Speed ctrl](#))
- The drive is stopped.

The results of the autotune routine are automatically transferred into parameters

- [23.01 Proport gain](#) (proportional gain of the speed controller)
- [23.02 Integration time](#) (integration time of the speed controller)
- [01.31 Mech time const](#) (mechanical time constant of the machinery).

The figure below shows speed responses at a speed reference step (typically 1...20%).



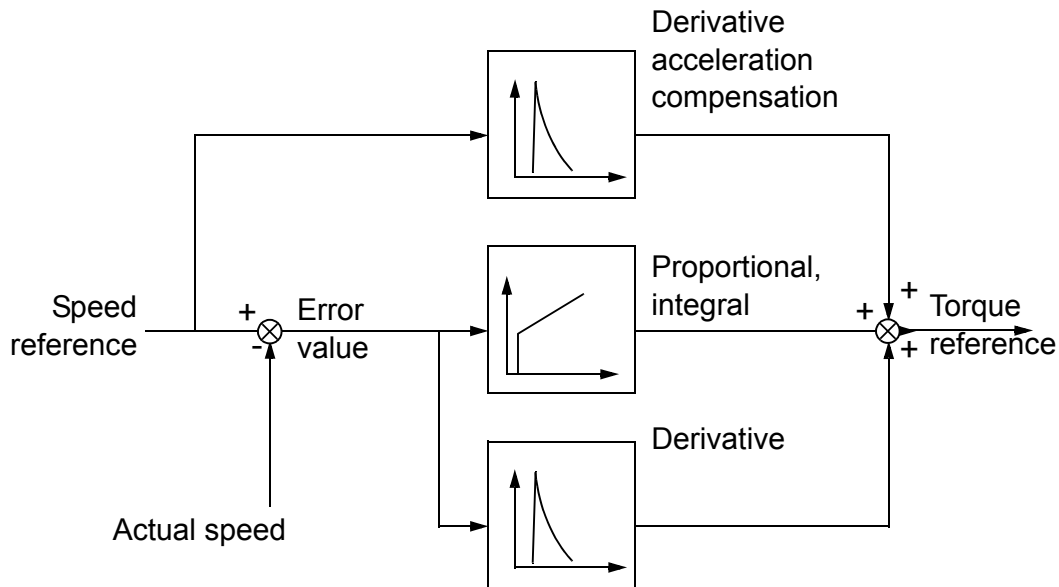
A: Undercompensated

B: Normally tuned (autotuning)

C: Normally tuned (manually). Better dynamic performance than with B

D: Overcompensated speed controller

The figure below is a simplified block diagram of the speed controller. The controller output is the reference for the torque controller.



Settings

Parameter group [23 Speed ctrl](#) (page [173](#)).

■ Encoder support

The program offers support for two encoders (or resolvers), encoder 1 and 2. Multiturn encoders are supported only as encoder 1. Three optional interface modules are available:

- TTL Encoder Interface FEN-01: two TTL inputs, TTL output (for encoder emulation and echo) and two digital inputs for position latching
- Absolute Encoder Interface FEN-11: absolute encoder input, TTL input, TTL output (for encoder emulation and echo) and two digital inputs for position latching
- Resolver Interface FEN-21: resolver input, TTL input, TTL output (for encoder emulation echo) and two digital inputs for position latching.
- HTL Encoder Interface FEN-31: HTL encoder input, TTL output (for encoder emulation and echo) and two digital inputs for position latching.

The interface module is connected to drive option Slot 1 or 2. **Note:** Two encoder interface modules of the same type are not allowed.

Settings

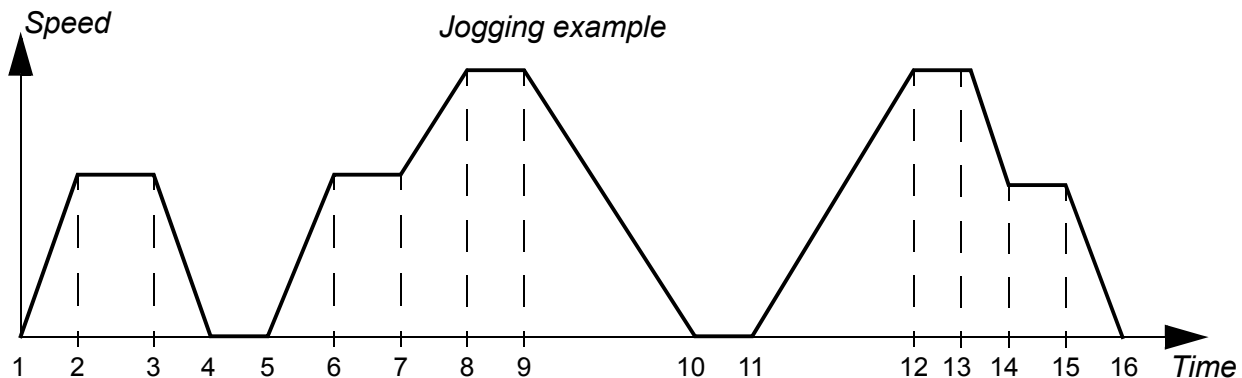
Parameter groups [91 Absol enc conf](#) (page [246](#)), [92 Resolver conf](#) (page [248](#)) and [93 Pulse enc conf](#) (page [249](#)).

■ Jogging

Two jogging functions (1 or 2) are available. When a jogging function is activated, the drive starts and accelerates to the defined jogging speed along the defined jogging acceleration ramp. When the function is deactivated, the drive decelerates to a stop along the defined jogging deceleration ramp. One push button can be used to start and stop the drive during jogging. The jogging function is typically used during servicing or commissioning to control the machinery locally.

Jogging functions 1 and 2 are activated by a parameter or through fieldbus. For activation through fieldbus, see parameter [02.22 FBA main cw](#) or [02.36 EFB main cw](#).

The figure and table below describe the operation of the drive during jogging. (Note that they cannot be directly applied to jogging commands through fieldbus as those require no enable signal; see parameter [10.09 Jog enable](#).) They also represent how the drive shifts to normal operation (= jogging inactive) when the drive start command is switched on. Jog cmd = State of the jogging input; Jog enable = Jogging enabled by the source set by parameter [10.09 Jog enable](#); Start cmd = State of the drive start command.



Phase	Jog cmd	Jog enable	Start cmd	Description
1-2	1	1	0	Drive accelerates to the jogging speed along the acceleration ramp of the jogging function.
2-3	1	1	0	Drive runs at the jogging speed.
3-4	0	1	0	Drive decelerates to zero speed along the deceleration ramp of the jogging function.
4-5	0	1	0	Drive is stopped.
5-6	1	1	0	Drive accelerates to the jogging speed along the acceleration ramp of the jogging function.
6-7	1	1	0	Drive runs at the jogging speed.
7-8	x	0	1	Jog enable is not active; normal operation continues.

Phase	Jog cmd	Jog enable	Start cmd	Description
8-9	x	0	1	Normal operation overrides the jogging. Drive follows the speed reference.
9-10	x	0	0	Drive decelerates to zero speed along the active deceleration ramp.
10-11	x	0	0	Drive is stopped.
11-12	x	0	1	Normal operation overrides the jogging. Drive accelerates to the speed reference along the active acceleration ramp.
12-13	1	1	1	Start command overrides the jog enable signal.
13-14	1	1	0	Drive decelerates to the jogging speed along the deceleration ramp of the jogging function.
14-15	1	1	0	Drive runs at the jogging speed.
15-16	x	0	0	Drive decelerates to zero speed along the deceleration ramp of the jogging function.

Note: Jogging is not operational when the drive start command is on, or if the drive is in local control.

Note: The ramp shape time is set to zero during jogging.

■ Scalar motor control

It is possible to select scalar control as the motor control method instead of Direct Torque Control (DTC). In scalar control mode, the drive is controlled with a frequency reference. However, the outstanding performance of DTC is not achieved in scalar control.

It is recommended to activate the scalar motor control mode in the following situations:

- In multimotor drives: 1) if the load is not equally shared between the motors, 2) if the motors are of different sizes, or 3) if the motors are going to be changed after motor identification (ID run)
- If the nominal current of the motor is less than 1/6 of the nominal output current of the drive
- If the drive is used without a motor connected (for example, for test purposes)
- If the drive runs a medium-voltage motor through a step-up transformer.

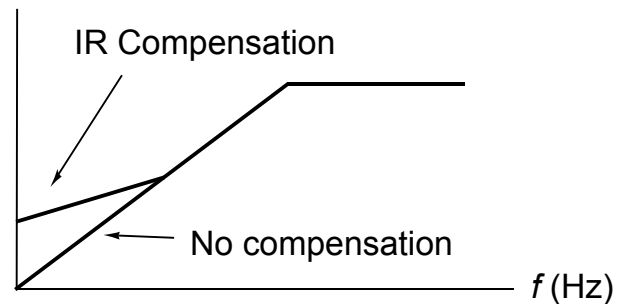
In scalar control, some standard features are not available.

IR compensation for a scalar controlled drive

IR compensation is active only when the motor control mode is scalar. When IR compensation is activated, the drive gives an extra voltage boost to the motor at low speeds. IR compensation is useful in applications that require a high break-away torque.

In Direct Torque Control (DTC), no IR compensation is possible or needed.

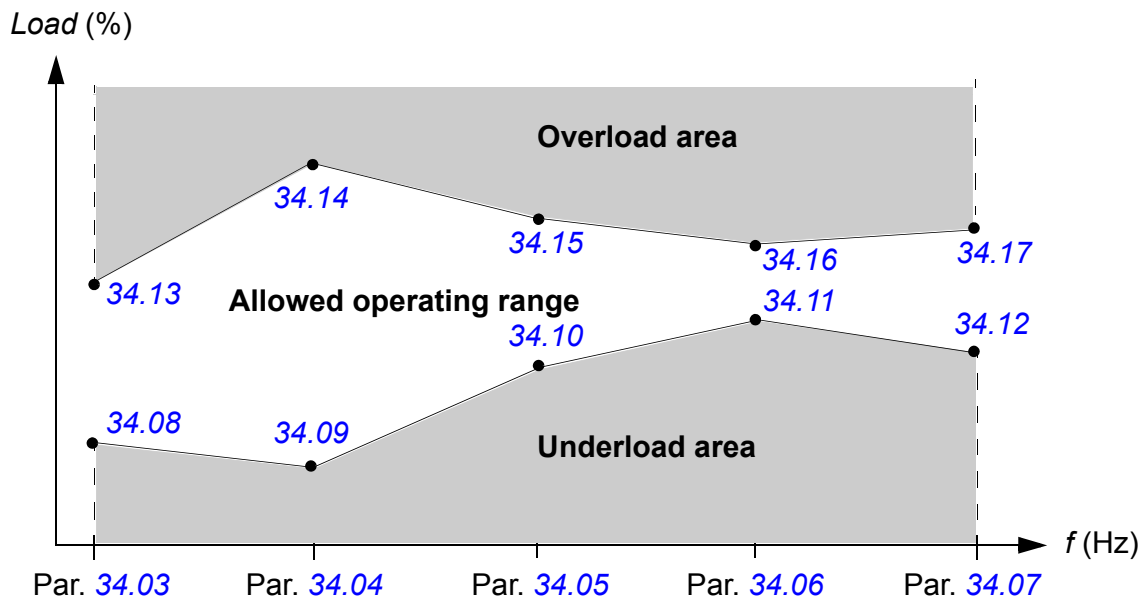
Motor Voltage



■ User-definable load curve

The drive output can be limited by defining a user-definable load curve. In practice, the user load curve consists of an overload and an underload curve, even though neither is compulsory. Each curve is formed by five points that represent output current or torque as a function of frequency.

An alarm or fault can be set up to occur when the curve is exceeded. The upper boundary (overload curve) can also be used as a torque or current limiter.

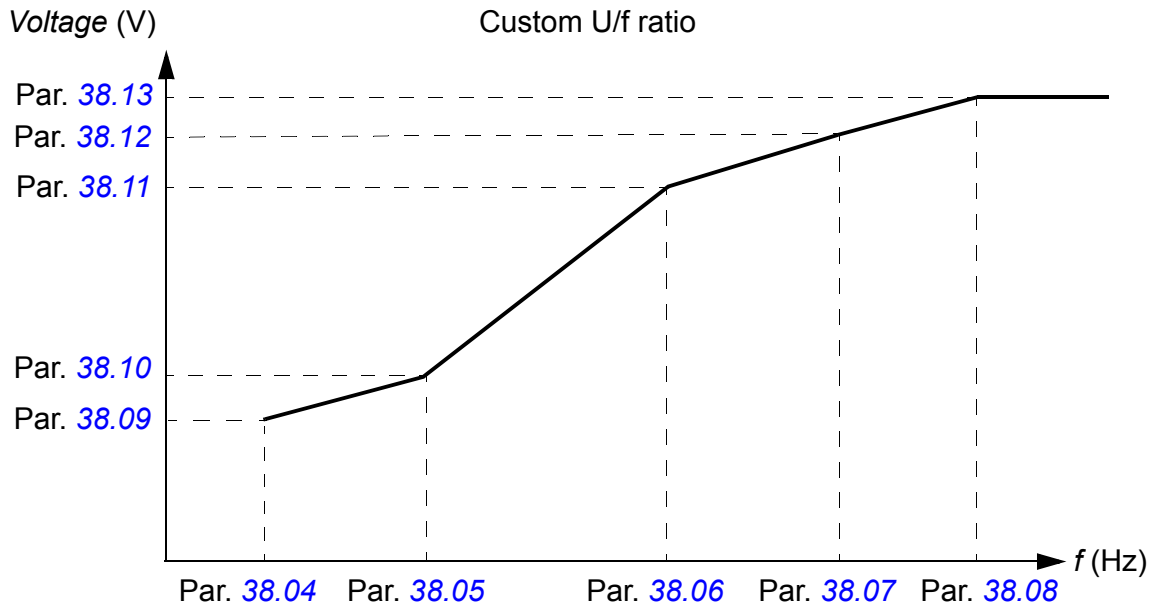


Settings

Parameter group [34 User load curve](#) (page 201).

■ User-definable U/f curve

The user can define a custom U/f curve (output voltage as a function of frequency). The curve can be used in special applications where linear and quadratic U/f ratios are not adequate (e.g. when motor break-away torque needs to be boosted).



Note: The U/f curve can be used in scalar control only, i.e., when [99.05 Motor ctrl mode](#) setting is *Scalar*.

Note: Each user-defined point must have a higher frequency and higher voltage than the previous point.



WARNING! High voltage at low frequencies may result in poor performance or motor damage due to overheating.

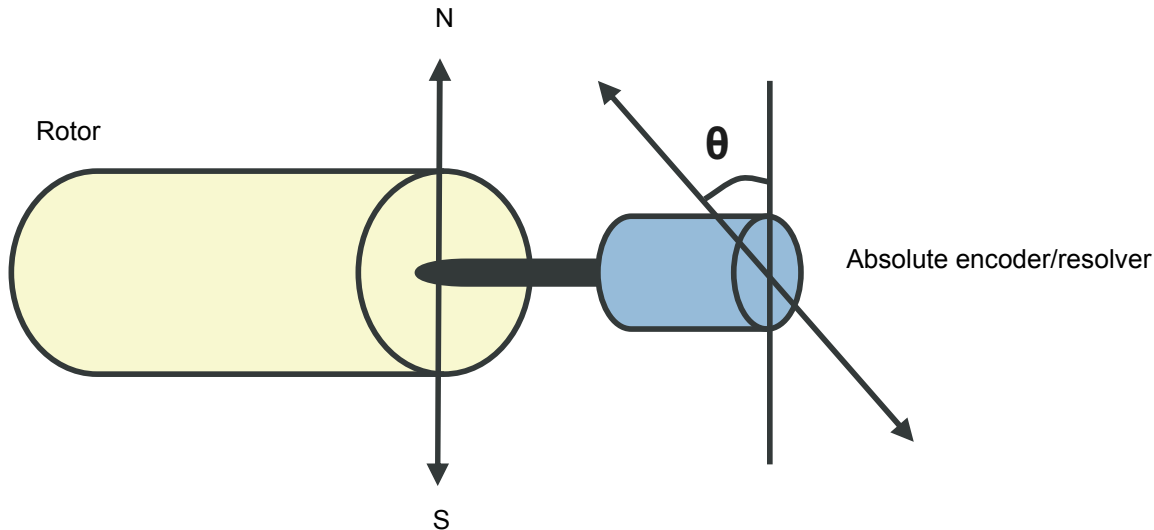
Settings

Parameter group [38 Flux ref](#) (page 214).

■ Autophasing

Autophasing is an automatic measurement routine to determine the angular position of the magnetic flux of a permanent magnet synchronous motor. The motor control requires the absolute position of the rotor flux in order to control motor torque accurately.

Sensors like absolute encoders and resolvers indicate the rotor position at all times after the offset between the zero angle of rotor and that of the sensor has been established. On the other hand, a standard pulse encoder determines the rotor position when it rotates but the initial position is not known. However, a pulse encoder can be used as an absolute encoder if it is equipped with Hall sensors, albeit with coarse initial position accuracy. The Hall sensors generate so-called commutation pulses that change their state six times during one revolution, so it is only known within which 60° sector of a complete revolution the initial position is.



The autophasing routine is performed with permanent magnet synchronous motors in the following cases:

1. One-time measurement of the rotor and encoder position difference when an absolute encoder, a resolver, or an encoder with commutation signals is used
2. At every power-up when an incremental encoder is used
3. With open-loop motor control, repetitive measurement of the rotor position at every start.

Several autophasing modes are available (see parameter [11.07 Autophasing mode](#)).

The turning mode is recommended especially with case 1 as it is the most robust and accurate method. In turning mode, the motor shaft is turned back and forward ($\pm 360/\text{polepairs}$)° in order to determine the rotor position. In case 3 (open-loop control), the shaft is turned only in one direction and the angle is smaller.

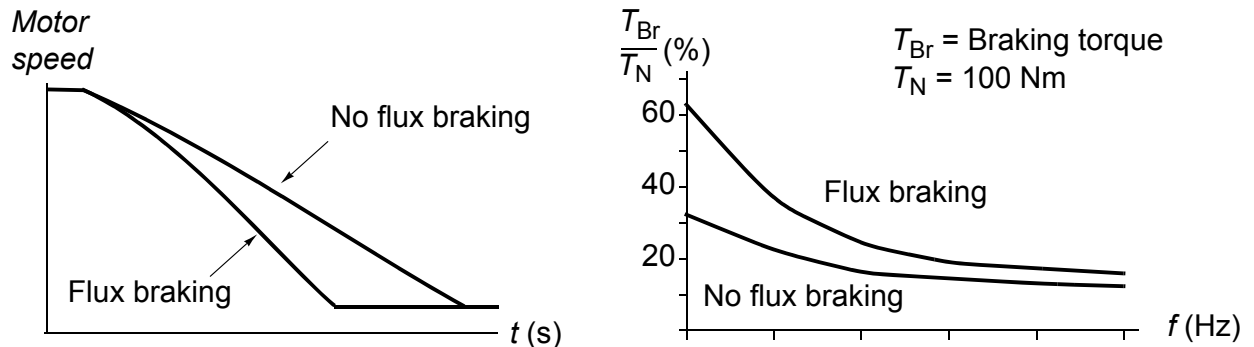
The standstill modes can be used if the motor cannot be turned (for example, when the load is connected). As the characteristics of motors and loads differ, testing must be done to find out the most suitable standstill mode.

A rotor position offset used in motor control can also be given by the user. See parameter [97.20 PM angle offset](#).

The drive is capable of determining the rotor position when started to a running motor in open-loop or closed-loop modes. In this situation, the setting of [11.07 Autophasing mode](#) has no effect.

■ Flux braking

The drive can provide greater deceleration by raising the level of magnetization in the motor. By increasing the motor flux, the energy generated by the motor during braking can be converted to motor thermal energy.



The drive monitors the motor status continuously, also during flux braking. Therefore, flux braking can be used both for stopping the motor and for changing the speed. The other benefits of flux braking are:

- The braking starts immediately after a stop command is given. The function does not need to wait for the flux reduction before it can start the braking.
- The cooling of the induction motor is efficient. The stator current of the motor increases during flux braking, not the rotor current. The stator cools much more efficiently than the rotor.
- Flux braking can be used with induction motors and permanent magnet synchronous motors.

Two braking power levels are available:

- Moderate braking provides faster deceleration compared to a situation where flux braking is disabled. The flux level of the motor is limited to prevent excessive heating of the motor.
- Full braking exploits almost all available current to convert the mechanical braking energy to motor thermal energy. Braking time is shorter compared to moderate braking. In cyclic use, motor heating may be significant.

Settings

Parameter [40.10 Flux braking](#) (page [217](#))

Application control

■ Application macros

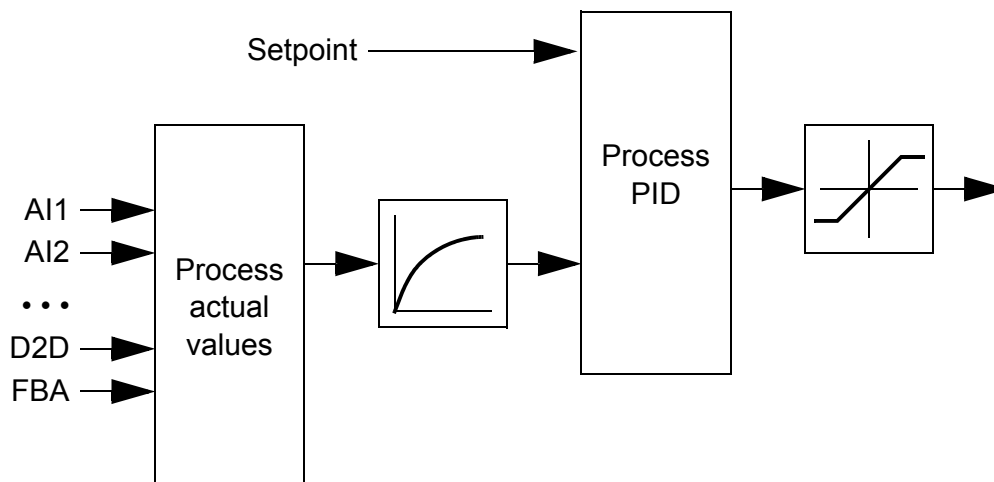
See chapter [Application macros](#) (page 89).

■ Process PID control

There is a built-in PID controller in the drive. The controller can be used to control process variables such as pressure, flow or fluid level.

In process PID control, a process reference (setpoint) is connected to the drive instead of a speed reference. An actual value (process feedback) is also brought back to the drive. The process PID control adjusts the drive speed in order to keep the measured process quantity (actual value) at the desired level (setpoint).

The simplified block diagram below illustrates the process PID control.



For a more detailed block diagram, see page [356](#).

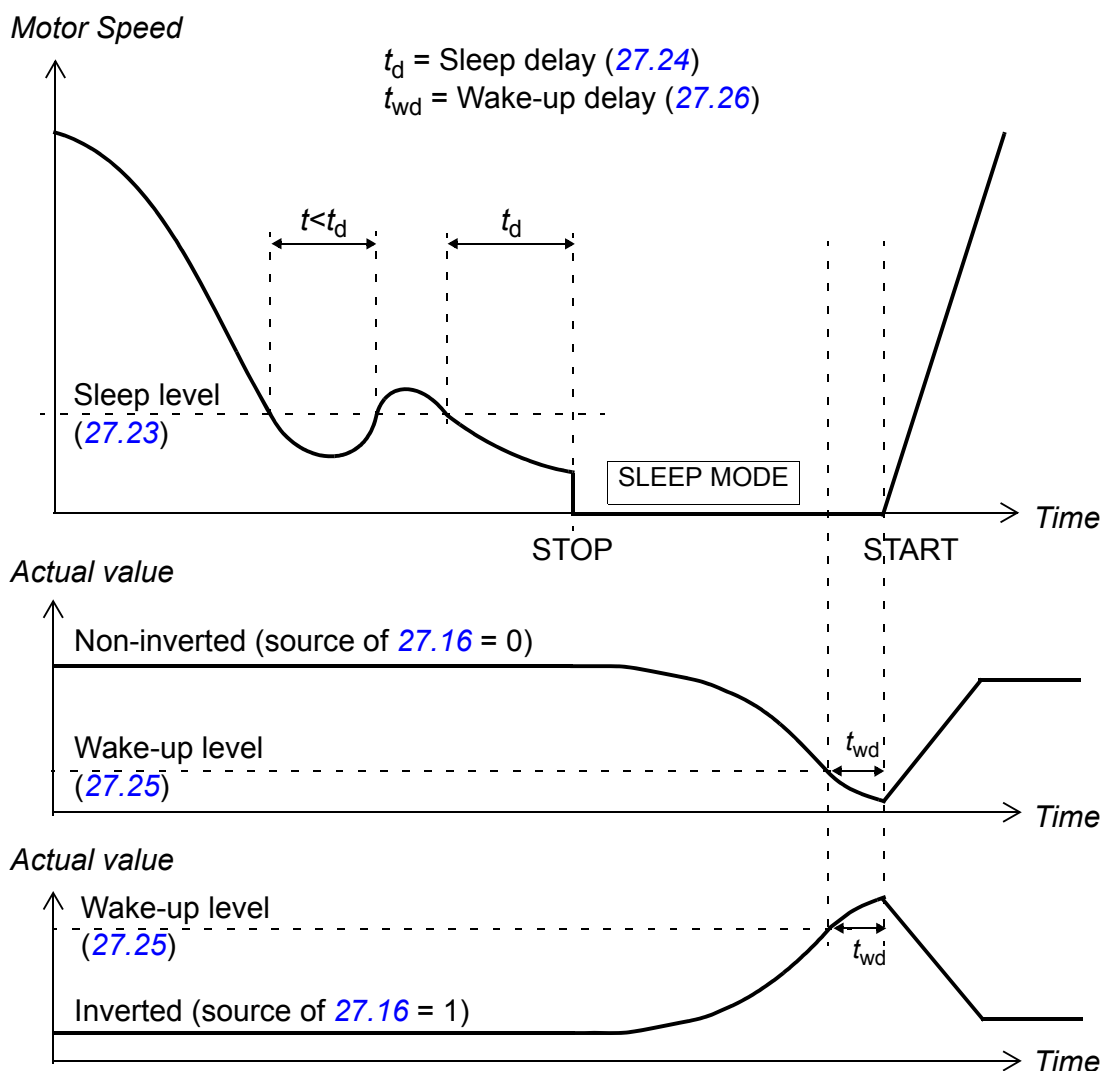
Quick configuration of the process PID controller

1. Select a setpoint source ([27.01 PID setpoint sel](#)).
2. Select a feedback source and set its minimum and maximum levels ([27.03 PID fbk1 src](#), [27.05 PID fbk1 max](#), [27.06 PID fbk1 min](#)). If a second feedback source is used, also set parameters [27.02 PID fbk func](#), [27.04 PID fbk2 src](#), [27.07 PID fbk2 max](#) and [27.08 PID fbk2 min](#).
3. Set the gain, integration time, derivation time, and the PID output levels ([27.12 PID gain](#), [27.13 PID integ time](#), [27.14 PID deriv time](#), [27.18 PID maximum](#) and [27.19 PID minimum](#)).
4. PID controller output is shown by parameter [04.05 Process PID out](#). Select it as the source of, for example, [21.01 Speed ref1 sel](#) or [24.01 Torq ref1 sel](#).

Sleep function for process PID control

The following example visualizes the operation of the sleep function.

The drive controls a pressure boost pump. The water consumption falls at night. As a consequence, the process PID controller decreases the motor speed. However, due to natural losses in the pipes and the low efficiency of the centrifugal pump at low speeds, the motor would never stop rotating. The sleep function detects the slow rotation and stops the unnecessary pumping after the sleep delay has passed. The drive shifts into sleep mode, still monitoring the pressure. The pumping resumes when the pressure falls under the predefined minimum level and the wake-up delay has passed.



Settings

Parameter group [27 Process PID](#) (page 185) and parameter [23.08 Speed additive](#) (page 177).

The PID control macro can be activated from the control panel main menu by selecting ASSISTANTS – Firmware assistants – Application Macro – PID control. See also page 94.

■ Mechanical brake control

A mechanical brake can be used for holding the motor and driven machinery at zero speed when the drive is stopped, or not powered.

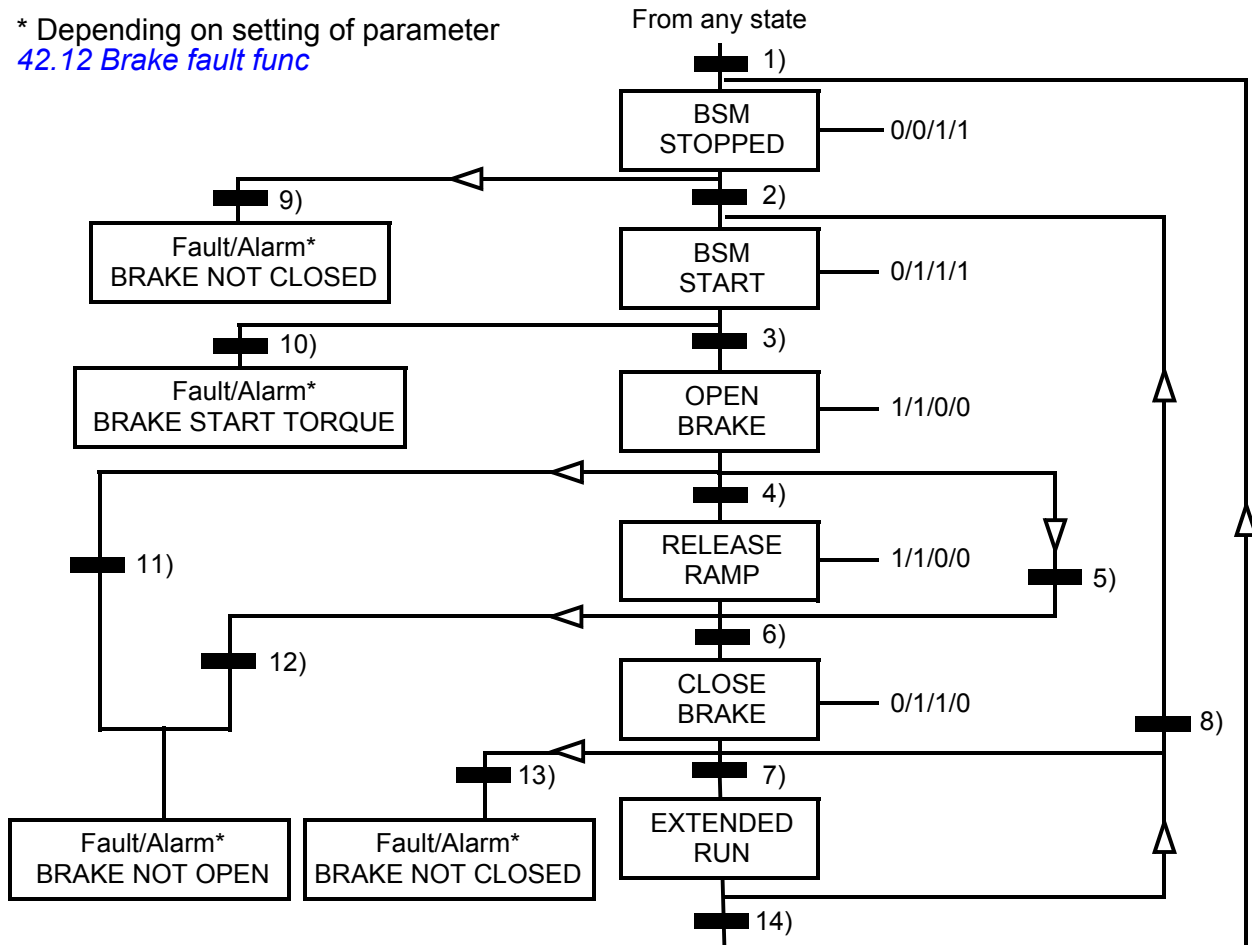
Parameters [03.15 Brake torq mem](#) and [03.16 Brake command](#) show the torque value stored when the brake close command is issued and the value of the brake command respectively.

Settings

Parameter group [42 Mech brake ctrl](#) (page 217).

BSM = Brake State Machine

* Depending on setting of parameter [42.12 Brake fault func](#)



State (Symbol

NN

 — W/X/Y/Z)

- NN: State name

- W/X/Y/Z: State outputs/operations

W: 1 = Brake open command is active. 0 = Brake close command is active. (Controlled through selected digital/relay output with signal [03.16 Brake command](#).)

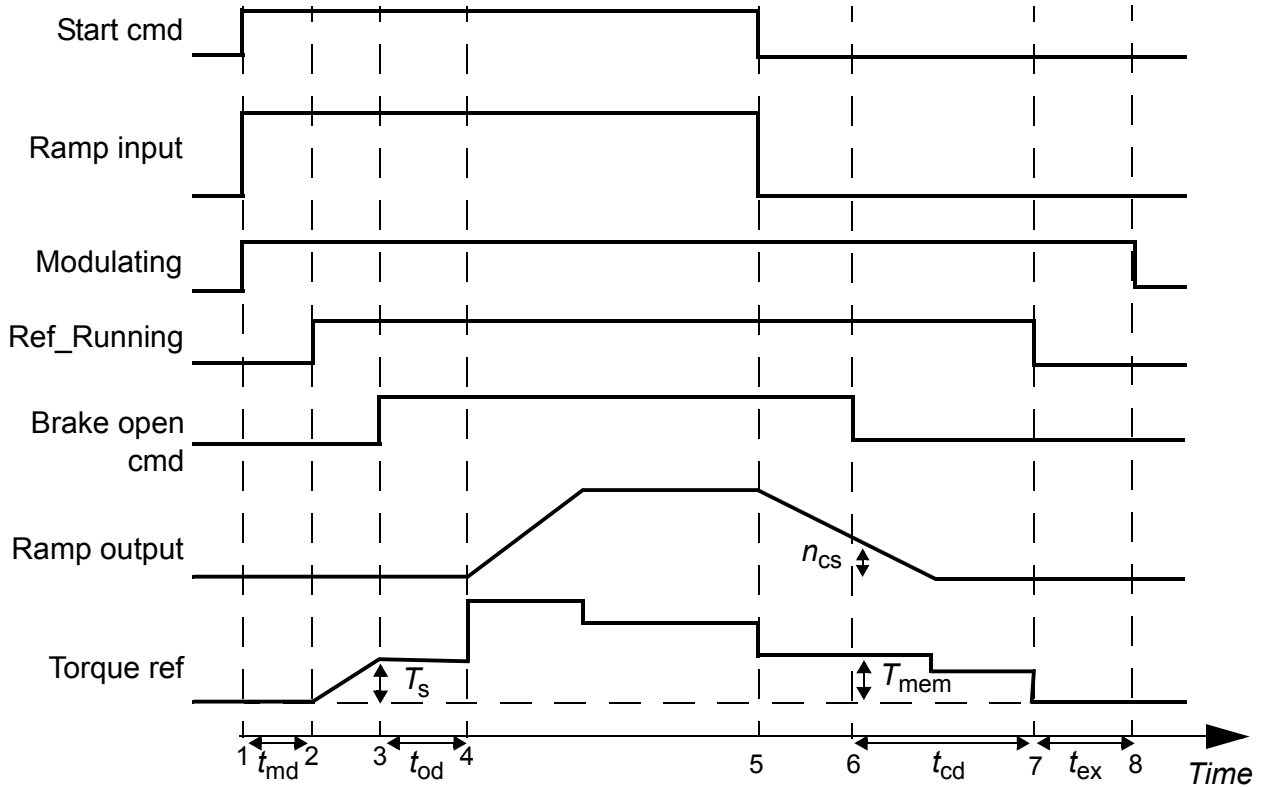
- X: 1 = Forced start (inverter is modulating). The function keeps the internal start command on until the brake is closed in spite of the status of the external stop command. Effective only when ramp stop has been selected as the stop mode ([11.03 Stop mode](#)). Run enable and faults override the forced start. 0 = No forced start (normal operation).
- Y: 1 = Drive control mode is forced to speed/scalar.
- Z: 1 = Ramp generator output is forced to zero. 0 = Ramp generator output is enabled (normal operation).

State change conditions (Symbol)

- 1) Brake control is active ([42.01 Brake ctrl](#) = *With ack* or *No ack*) OR modulation of the drive is requested to stop. The drive control mode is forced to speed/scalar.
 - 2) External start command is on AND brake open request is on (source selected by [42.10 Brake close req](#) is 0) AND reopen delay ([42.07 Reopen delay](#)) has elapsed.
 - 3) Starting torque required at brake release is reached ([42.08 Brake open torq](#)) AND brake hold is not active ([42.11 Brake hold open](#)). **Note:** With scalar control, the defined starting torque has no effect.
 - 4) Brake is open (acknowledgement source selected by par. [42.02 Brake acknowl](#) is 1) AND the brake open delay has elapsed ([42.03 Open delay](#)). Start = 1.
 - 5) 6) Start = 0 OR brake close command is active AND actual motor speed < brake close speed ([42.05 Close speed](#)) AND close command delay ([42.06 Close cmd delay](#)) has elapsed.
 - 7) Brake is closed (acknowledgement = 0) AND brake close delay ([42.04 Close delay](#)) has elapsed. Start = 0.
 - 8) Start = 1 AND brake open request is on (source selected by [42.10 Brake close req](#) is 0) AND reopen delay has elapsed.
 - 9) Brake is open (acknowledgement = 1) AND brake close delay has elapsed.
 - 10) Defined starting torque at brake release is not reached.
 - 11) Brake is closed (acknowledgement = 0) AND brake open delay has elapsed.
 - 12) Brake is closed (acknowledgement = 0).
 - 13) Brake is open (acknowledgement = 1) AND brake close delay has elapsed. Fault is generated after brake close fault delay ([42.13 Close flt delay](#)) has elapsed.
 - 14) Brake is closed (acknowledgement = 1) AND extended run delay ([42.14 Extend run time](#)) has elapsed. Start = 0.
-

Operation time scheme

The simplified time scheme below illustrates the operation of the brake control function.



- T_s Start torque at brake release (parameter [42.08 Brake open torq](#))
- T_{mem} Stored torque value at brake close (signal [03.15 Brake torq mem](#))
- t_{md} Motor magnetising delay
- t_{od} Brake open delay (parameter [42.03 Open delay](#))
- n_{cs} Brake close speed (parameter [42.05 Close speed](#))
- t_{ccd} Brake close command delay (parameter [42.06 Close cmd delay](#))
- t_{cd} Brake close delay (parameter [42.04 Close delay](#))
- t_{ex} Extended run time

Example

The figure below shows a brake control application example.

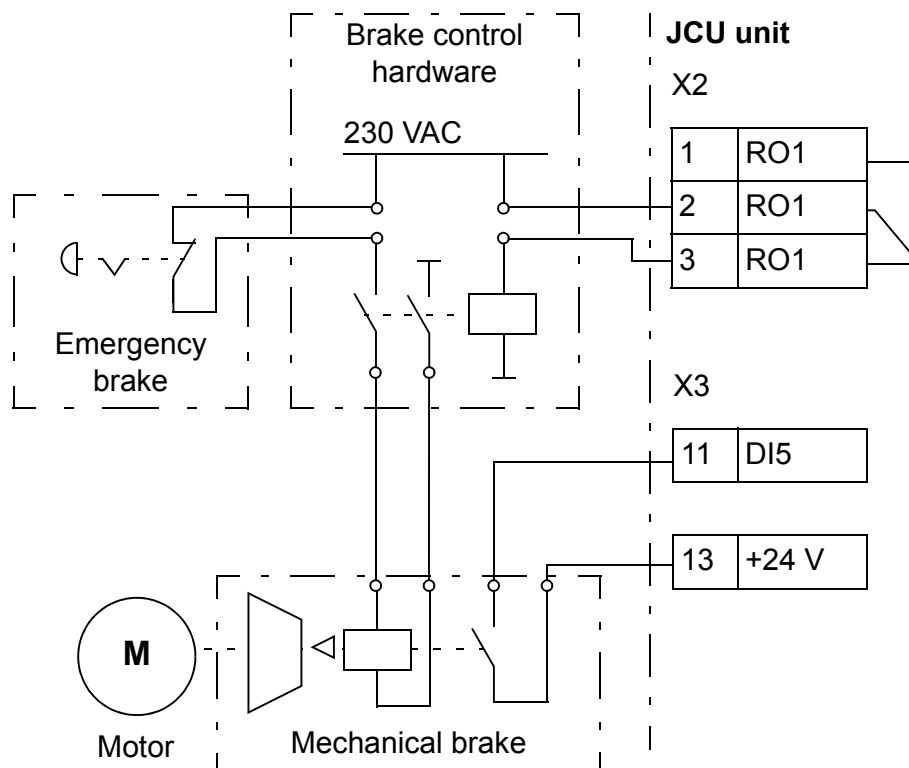


WARNING! Make sure that the machinery into which the drive with brake control function is integrated fulfils the personnel safety regulations. Note that the frequency converter (a Complete Drive Module or a Basic Drive Module, as defined in IEC 61800-2), is not considered as a safety device mentioned in the European Machinery Directive and related harmonised standards. Thus, the personnel safety of the complete machinery must not be based on a specific frequency converter feature (such as the brake control function), but it has to be implemented as defined in the application specific regulations.

The brake on/off is controlled via signal [03.16 Brake command](#). The source for the brake supervision is selected by parameter [42.02 Brake acknowl](#).

The brake control hardware and wirings need to be done by the user.

- Brake on/off control through selected relay/digital output.
- Brake supervision through selected digital input.
- Emergency brake switch in the brake control circuit.
- Brake on/off control through relay output (i.e. parameter [14.42 RO1 src](#) setting is P.03.16.00 = [03.16 Brake command](#)).
- Brake supervision through digital input DI5 (i.e. parameter [42.02 Brake acknowl](#) setting is P.02.01.04 = [02.01 DI status](#), bit 4)



■ Timers

It is possible to define four different daily or weekly time periods. The time periods can be used to control four different timers. The on/off statuses of the four timers are indicated by bits 0...3 of parameter *06.14 Timed func stat*, from where the signal can be connected to any parameter with a bit pointer setting (see page 102). In addition, bit 4 of parameter *06.14* is on if any one of the four timers is on.

Each time period can be assigned to multiple timers; likewise, a timer can be controlled by multiple time periods.

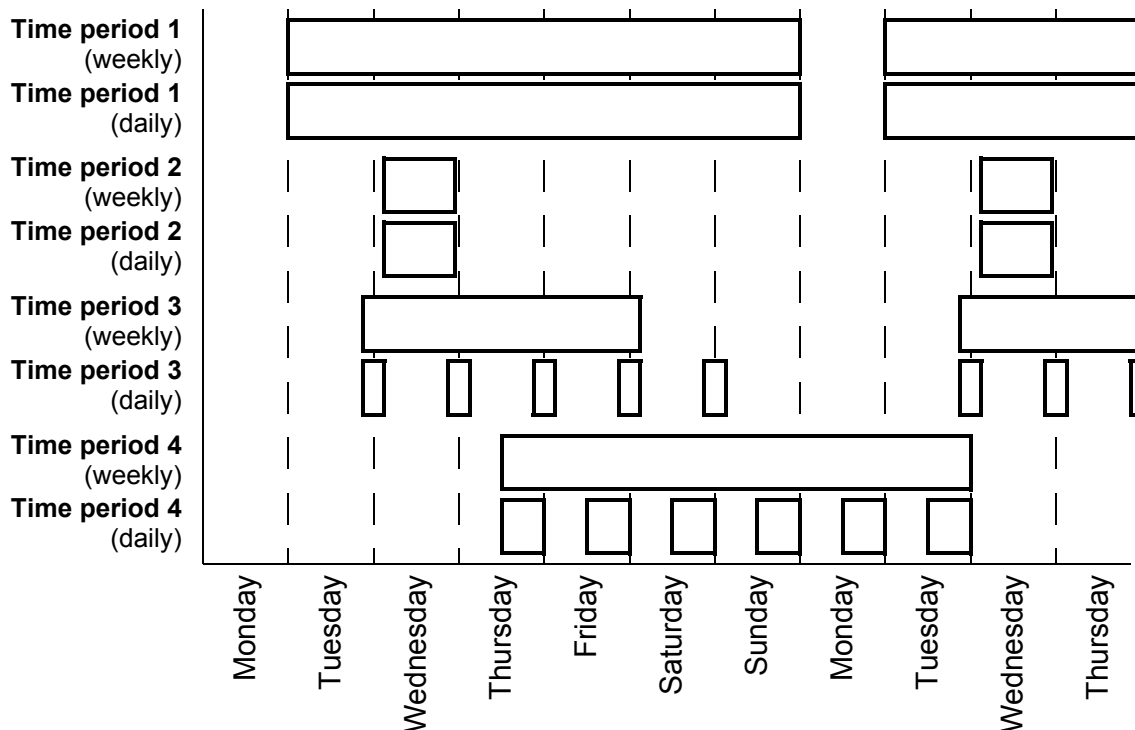
The figure below presents how different time periods are active in daily and weekly modes.

Time period 1: Start time 00:00:00; Stop time 00:00:00 or 24:00:00; Start on Tuesday; Stop day Sunday

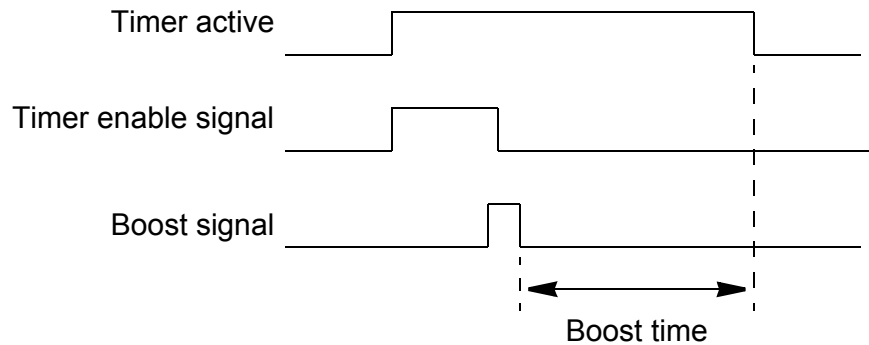
Time period 2: Start time 03:00:00; Stop time 23:00:00; Start day Wednesday; Stop day Wednesday

Time period 3: Start time 21:00:00; Stop time 03:00:00; Start day Tuesday; Stop day Saturday

Time period 4: Start time 12:00:00; Stop time 00:00:00 or 24:00:00; Start day Thursday; Stop day Tuesday



A “boost” function is also available for the activation of the timers: a signal source can be selected to extend the activation time for a parameter-adjustable time period.



Settings

Parameter group [36 Timed functions](#) (page [209](#)).

DC voltage control

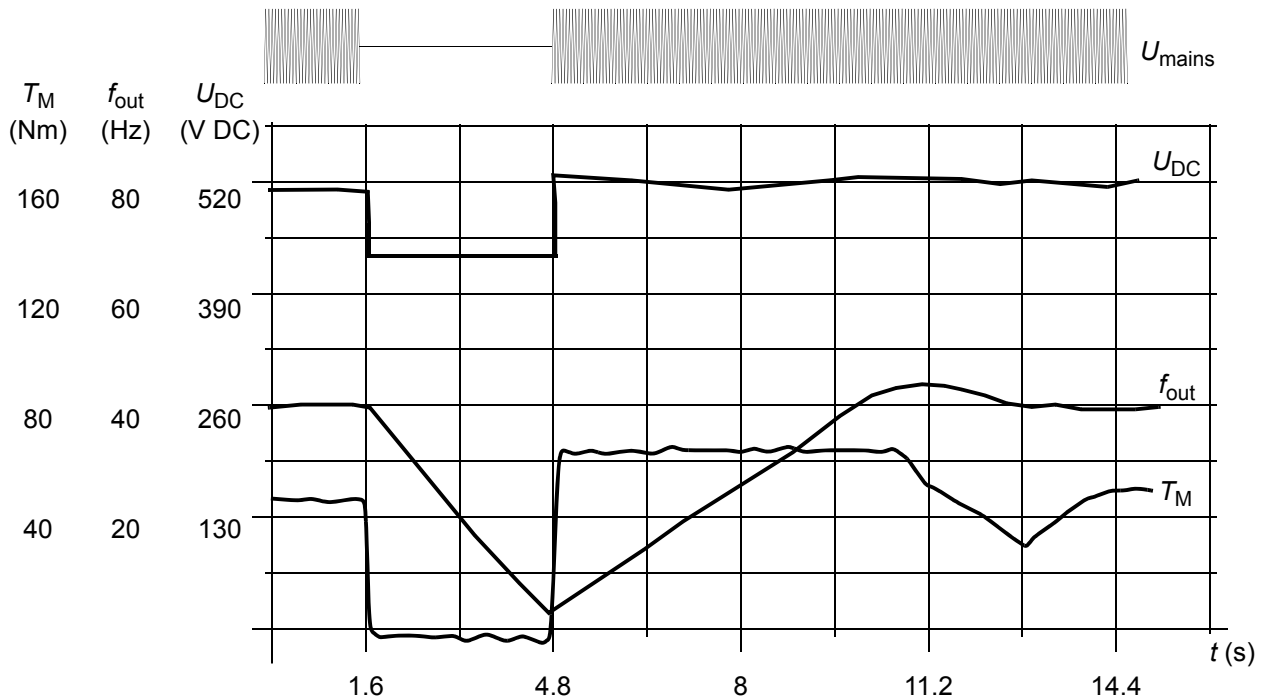
■ Overvoltage control

Overvoltage control of the intermediate DC link is needed with two-quadrant line-side converters when the motor operates within the generating quadrant. To prevent the DC voltage from exceeding the overvoltage control limit, the overvoltage controller automatically decreases the generating torque when the limit is reached.

■ Undervoltage control

If the incoming supply voltage is cut off, the drive will continue to operate by utilizing the kinetic energy of the rotating motor. The drive will be fully operational as long as the motor rotates and generates energy to the drive. The drive can continue the operation after the break if the main contactor remained closed.

Note: Units equipped with a main contactor must be equipped with a hold circuit (e.g. UPS) to keep the contactor control circuit closed during a short supply break.



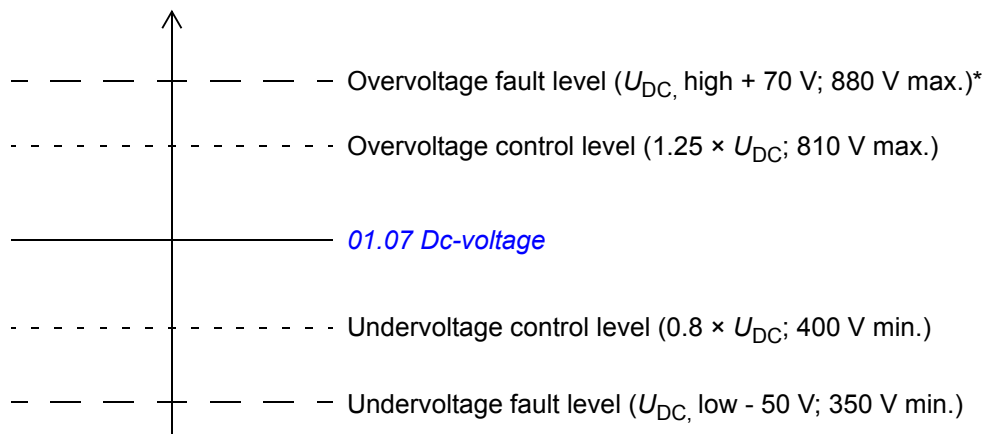
U_{DC} = intermediate circuit voltage of the drive, f_{out} = output frequency of the drive, T_M = motor torque

Loss of supply voltage at nominal load ($f_{out} = 40$ Hz). The intermediate circuit DC voltage drops to the minimum limit. The controller keeps the voltage steady as long as the mains is switched off. The drive runs the motor in generator mode. The motor speed falls but the drive is operational as long as the motor has enough kinetic energy.

■ Voltage control and trip limits

The control and trip limits of the intermediate DC voltage regulator are relative either to a supply voltage value provided by the user, or to an automatically-determined supply voltage. The actual voltage used is shown by parameter [01.19 Used supply volt](#). The DC voltage (U_{DC}) equals 1.35 times this value.

Automatic identification of the supply voltage is performed every time the drive is powered on. Automatic identification can be disabled by parameter [47.03 SupplyVoltAutold](#); the user can then define the voltage manually at parameter [47.04 Supply voltage](#).



$$U_{DC} = 1.35 \times 01.19 \text{ Used supply volt}$$

$$U_{DC, \text{ high}} = 1.25 \times U_{DC}$$

$$U_{DC, \text{ low}} = 0.8 \times U_{DC}$$

*Drives with 230 V supply voltage (ACS850-04-xxxx-2): The overvoltage fault level is set to 500 V and the minimum levels for undervoltage control and fault are removed.

The intermediate DC circuit is charged over an internal resistor which is bypassed when the capacitors are considered charged and the voltage has stabilized.

Settings

Parameter group [47 Voltage ctrl](#) (page [228](#)).

■ Brake chopper

The built-in brake chopper of the drive can be used to handle the energy generated by a decelerating motor.

When the brake chopper is enabled and a resistor connected, the chopper will start conducting when the DC link voltage of the drive reaches $U_{DC_BR} - 30$ V. The maximum braking power is achieved at $U_{DC_BR} + 30$ V.

$$U_{DC_BR} = 1.35 \times 1.25 \times 01.19 \text{ Used supply volt.}$$

Settings

Parameter group [48 Brake chopper](#) (page [228](#)).

Safety and protections

■ Emergency stop

Note: The user is responsible for installing the emergency stop devices and all the additional devices needed for the emergency stop to fulfil the required emergency stop category classes. For more information, contact your local ABB representative.

The emergency stop signal is to be connected to the digital input which is selected as the source for the emergency stop activation (par. [10.13 Em stop off3](#) or [10.15 Em stop off1](#)). Emergency stop can also be activated through fieldbus ([02.22 FBA main cw](#) or [02.36 EFB main cw](#)).

Note: When an emergency stop signal is detected, the emergency stop function cannot be cancelled even though the signal is cancelled.

■ Thermal motor protection

The motor can be protected against overheating by

- the motor thermal protection model
- measuring the motor temperature with PTC, Pt100 or KTY84 sensors. This will result in a more accurate motor model.

Thermal motor protection model

The drive calculates the temperature of the motor on the basis of the following assumptions:

1) When power is applied to the drive for the first time, the motor is at ambient temperature (defined by parameter [31.09 Mot ambient temp](#)). After this, when power is applied to the drive, the motor is assumed to be at the estimated temperature.

2) Motor temperature is calculated using the user-adjustable motor thermal time and motor load curve. The load curve should be adjusted in case the ambient temperature exceeds 30 °C.

It is possible to adjust the motor temperature supervision limits and select how the drive reacts when overtemperature is detected.

Note: The motor thermal model can be used when only one motor is connected to the inverter.

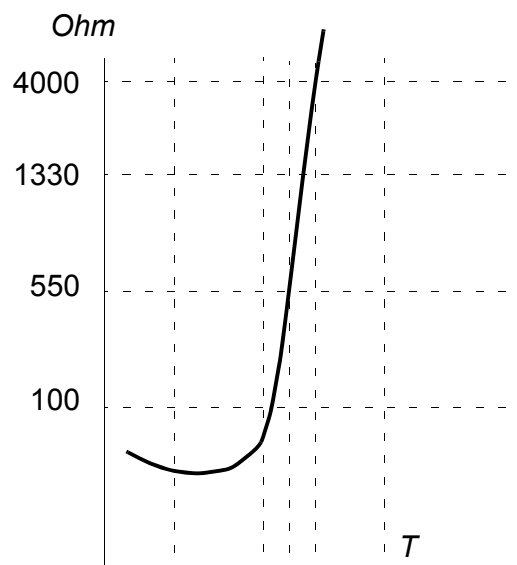
Temperature monitoring using PTC sensors

A PTC sensor can be connected between +24 V and digital input DI6 of the drive, or to an optional encoder interface module FEN-xx.

The resistance of the sensor increases as the motor temperature rises over the sensor reference temperature T_{ref} , as does the voltage over the resistor. The temperature measurement function reads the voltage and converts it into ohms.

The figure and table below show typical PTC sensor resistance values as a function of the motor operating temperature.

Temperature	PTC resistance
Normal	0...1 kohm
Excessive	≥ 1 kohm



For the wiring of the sensor, refer to the *Hardware Manual* of the drive.

Temperature monitoring using Pt100 sensors

A Pt100 sensor can be connected to AI1 and AO1 on the JCU Control Unit, or to the first available AI and AO on the optional FIO-xx I/O extension module.

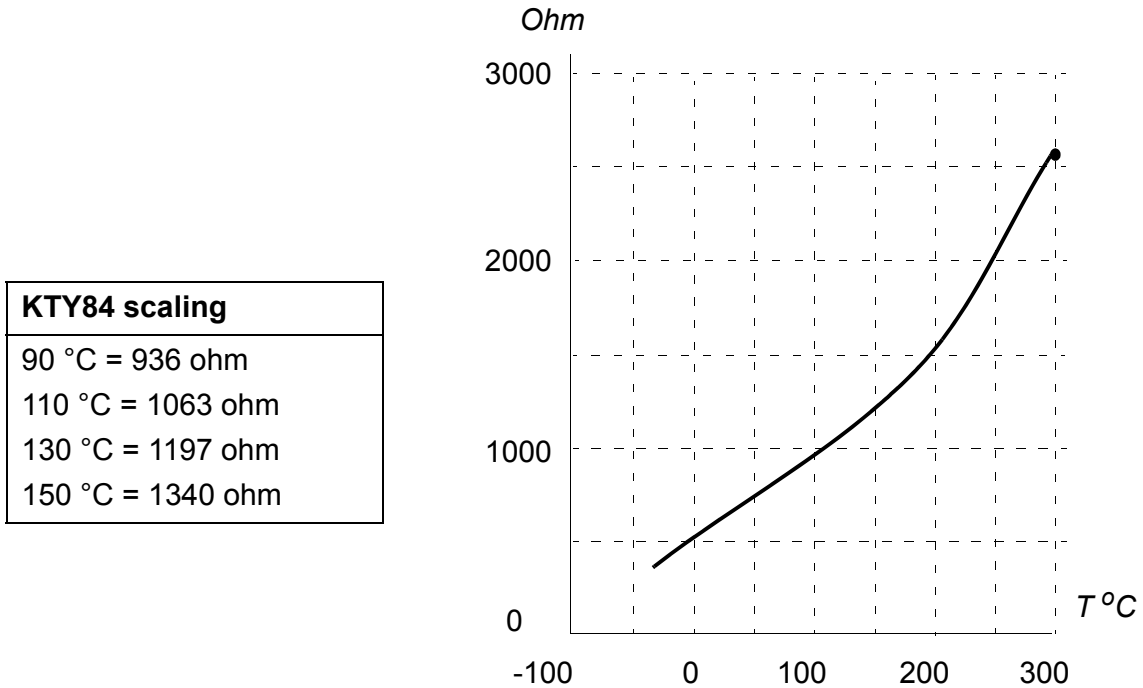
The analog output feeds constant current through the sensor. The sensor resistance increases as the motor temperature rises, as does the voltage over the sensor. The temperature measurement function reads the voltage through the analog input and converts it to degrees centigrade.

For the wiring of the sensor, refer to the *Hardware Manual* of the drive.

Temperature monitoring using KTY84 sensors

A KTY84 sensor can be connected to an optional encoder interface module FEN-xx.

The figure and table below show typical KTY84 sensor resistance values as a function of the motor operating temperature.



It is possible to adjust the motor temperature supervision limits and select how the drive reacts when overtemperature is detected.

For the wiring of the sensor, refer to the *Hardware Manual* of the drive.

Settings

Parameter group [31 Motor therm prot](#) (page [192](#)).

■ Programmable protection functions

Start interlock (parameter [10.20](#))

The parameter selects how the drive reacts to loss of start interlock signal (DIIL).

External fault (parameter [30.01](#))

A source for an external fault signal is selected by this parameter. When the signal is lost, a fault is generated.

Local control loss detection (parameter [30.03](#))

The parameter selects how the drive reacts to a control panel or PC tool communication break.

Motor phase loss detection (parameter [30.04](#))

The parameter selects how the drive reacts whenever a motor phase loss is detected.

Earth fault detection (parameter [30.05](#))

The earth fault detection function is based on sum current measurement. Note that

- an earth fault in the supply cable does not activate the protection
- in a grounded supply, the protection activates in 200 milliseconds
- in an ungrounded supply, the supply capacitance should be 1 microfarad or more
- the capacitive currents caused by shielded motor cables up to 300 metres will not activate the protection
- the protection is deactivated when the drive is stopped.

Supply phase loss detection (parameter [30.06](#))

The parameter selects how the drive reacts whenever a supply phase loss is detected.

Safe torque off detection (parameter [30.07](#))

The drive monitors the status of the Safe torque off input. For more information on the Safe torque off function, see the *Hardware Manual* of the drive, and *Application guide - Safe torque off function for ACSM1, ACS850 and ACQ810 drives* (3AFE68929814 [English]).

Switched supply and motor cabling (parameter [30.08](#))

The drive can detect if the supply and motor cables have accidentally been switched (for example, if the supply is connected to the motor connection of the drive). The parameter selects if a fault is generated or not.

Stall protection (parameters [30.09](#)...[30.12](#))

The drive protects the motor in a stall situation. It is possible to adjust the supervision limits (current, frequency and time) and choose how the drive reacts to a motor stall condition.

■ Automatic fault resets

The drive can automatically reset itself after overcurrent, overvoltage, undervoltage, external and “analog input below minimum” faults. By default, automatic resets are off and must be separately activated by the user.

Settings

Parameter group [32 Automatic reset](#) (page [197](#)).

Diagnostics

■ Signal supervision

Three signals can be selected to be supervised by this function. Whenever the signal exceeds (or falls below) a predefined limit, a bit of [06.13 Superv status](#) is activated. Absolute values can be used.

Settings

Parameter group [33 Supervision](#) (page [198](#)).

■ Maintenance counters

The program has six different maintenance counters that can be configured to generate an alarm when the counter reaches a pre-defined limit. The counter can be set to monitor any parameter. This feature is especially useful as a service reminder.

There are three types of counters:

- On-time counter. Measures the time a digital source (for example, a bit in a status word) is on.
- Rising edge counter. This counter is incremented whenever the monitored digital source changes state from 0 to 1.
- Value counter. This counter measures, by integration, the monitored parameter. An alarm is given when the calculated area below the signal peak exceeds a user-defined limit.

Settings

Parameter group [44 Maintenance](#) (page [221](#)).

■ Energy saving calculator

This feature consists of three functionalities:

- An energy optimizer that adjusts the motor flux in such a way that the total efficiency is maximized
- A counter that monitors used and saved energy by the motor and displays them in kWh, currency or volume of CO₂ emission, and
- A load analyzer showing the load profile of the drive (see page 85).

Note: The accuracy of the energy savings calculation is directly dependent on the accuracy of the reference motor power given in parameter [45.08 Reference power](#).

Settings

Parameter group [45 Energy optimising](#) (page 227).

■ Load analyzer

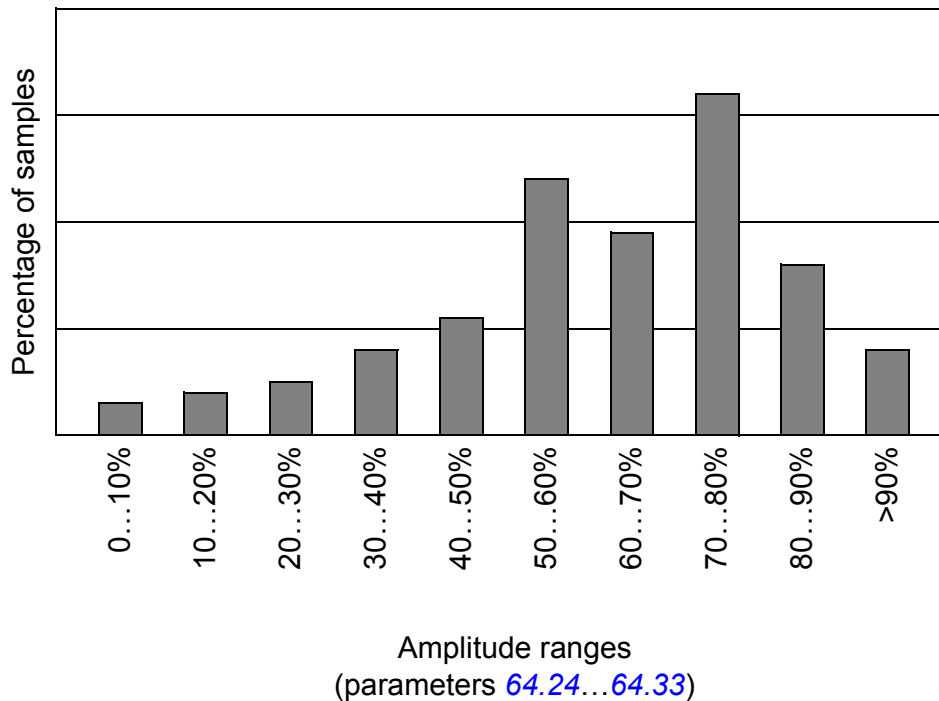
Peak value logger

The user can select a signal to be monitored by the peak value logger. The logger records the peak value of the signal along with the time the peak occurred, as well as motor current, DC voltage and motor speed at the time of the peak.

Amplitude loggers

The drive has two amplitude loggers.

For amplitude logger 2, the user can select a signal to be sampled at 200 ms intervals when the drive is running, and specify a value that corresponds to 100%. The collected samples are sorted into 10 read-only parameters according to their amplitude. Each parameter represents an amplitude range 10 percentage points wide, and displays the percentage of the collected samples that fall within that range.



Amplitude logger 1 is fixed to monitor motor current, and cannot be reset. With amplitude logger 1, 100% corresponds to the maximum output current of the drive (I_{Max}).

Settings

Parameter group [64 Load analyzer](#) (page [241](#)).

Miscellaneous

■ Backup and restore of drive contents

General

The drive offers a possibility of backing up numerous settings and configurations to external storage such as a PC file (using the DriveStudio tool) and the internal memory of the control panel. These settings and configurations can then be restored to the drive, or a number of drives.

Backup using DriveStudio includes

- Parameter settings
- User parameter sets
- Application program.

Backup using the drive control panel includes

- Parameter settings
- User parameter sets.

For detailed instructions for performing the backup/restore, refer to page [39](#) and the DriveStudio documentation.

Limitations

A backup can be done without interfering with drive operation, but restoring a backup always resets and reboots the control unit, so restore is not possible with the drive running.

Restoring backup files from one firmware version to another is considered risky, so the results should be carefully observed and verified when done for the first time. The parameters and application support are bound to change between firmware versions and backups are not always compatible with other firmware versions even if restore is allowed by the backup/restore tool. Before using the backup/restore functions between different firmware versions, refer to the release notes of each version.

Applications should not be transferred between different firmware versions. Contact the supplier of the application when it needs to be updated for a new firmware version.

Parameter restore

Parameters are divided into three different groups that can be restored together or individually:

- Motor configuration parameters and identification (ID) run results
- Fieldbus adapter and encoder settings
- Other parameters.

For example, retaining the existing ID run results in the drive will make a new ID run unnecessary.

Restore of individual parameters can fail for the following reasons:

- The restored value does not fall within the minimum and maximum limits of the drive parameter
- The type of the restored parameter is different from that in the drive
- The restored parameter does not exist in the drive (often the case when restoring the parameters of a new firmware version to a drive with an older version)
- The backup does not contain a value for the drive parameter (often the case when restoring the parameters of an old firmware version to a drive with a newer version).

In these cases, the parameter is not restored; the backup/restore tool will warn the user and offer a possibility to set the parameter manually.

User parameter sets

The drive has four user parameter sets that can be saved to the permanent memory and recalled using drive parameters. It is also possible to use digital inputs to switch between different user parameter sets. See the descriptions of parameters [16.09...16.12](#).

A user parameter set contains all values of parameter groups 10 to 99 (except the configuration settings for fieldbus adapter communication).

As the motor settings are included in the user parameter sets, make sure the settings correspond to the motor used in the application before recalling a user set. In an application where different motors are used with one drive, the motor ID run needs to be performed with each motor and saved to different user sets. The appropriate set can then be recalled when the motor is switched.

Settings

Parameter group [16 System](#) (page [160](#)).

■ Data storage parameters

Four 16-bit and four 32-bit parameters are reserved for data storage. These parameters are unconnected and can be used for linking, testing and commissioning purposes. They can be written to and read from using other parameters' pointer settings.

Settings

Parameter group [49 Data storage](#) (page [229](#)).

■ Drive-to-drive link

The drive-to-drive link is a daisy-chained RS-485 transmission line that allows basic master/follower communication with one master drive and multiple followers.

See chapter [Drive-to-drive link](#) (page [341](#)).

Settings

Parameter group [57 D2D communication](#) (page [235](#)).



Application macros

efesotomasyon.com

What this chapter contains

This chapter describes the intended use, operation and default control connections of the application macros.

More information on the connectivity of the JCU control unit is given in the *Hardware Manual* of the drive.

General

Application macros are pre-defined parameter sets. When starting up the drive, the user typically selects one of the macros as a basis, makes the essential changes and saves the result as a user parameter set.

Application macros are activated through the control panel main menu by selecting ASSISTANTS – Application Macro. User parameter sets are managed by the parameters in group [16 System](#).

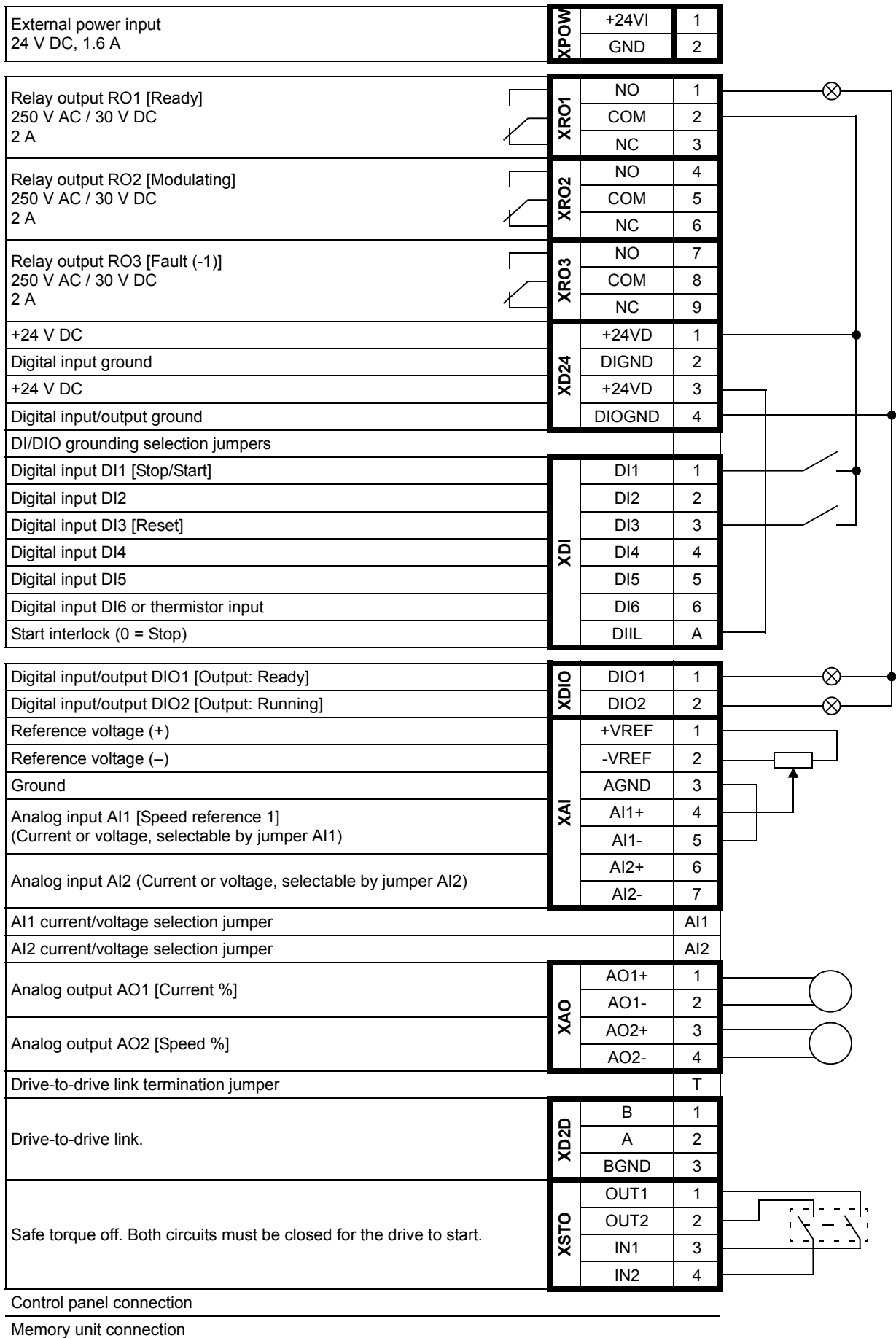
Factory macro

The Factory macro is suited to relatively straightforward speed control applications such as conveyors, pumps and fans, and test benches.

In external control, the control location is EXT1. The drive is speed-controlled; the reference signal is connected to analog input AI1. The sign of the reference determines the running direction. The start/stop commands are given through digital input DI1. Faults are reset through DI3.

The default parameter settings for the Factory macro are listed in chapter [Additional parameter data](#) (page 257).

Default control connections for the Factory macro



Hand/Auto macro

The Hand/Auto macro is suited for speed control applications where two external control devices are used.

The drive is speed-controlled from the external control locations EXT1 and EXT2. The selection between the control locations is done through digital input DI3.

The start/stop signal for EXT1 is connected to DI1 while running direction is determined by DI2. For EXT2, start/stop commands are given through DI6, the direction through DI5.

The reference signals for EXT1 and EXT2 are connected to analog inputs AI1 and AI2 respectively.

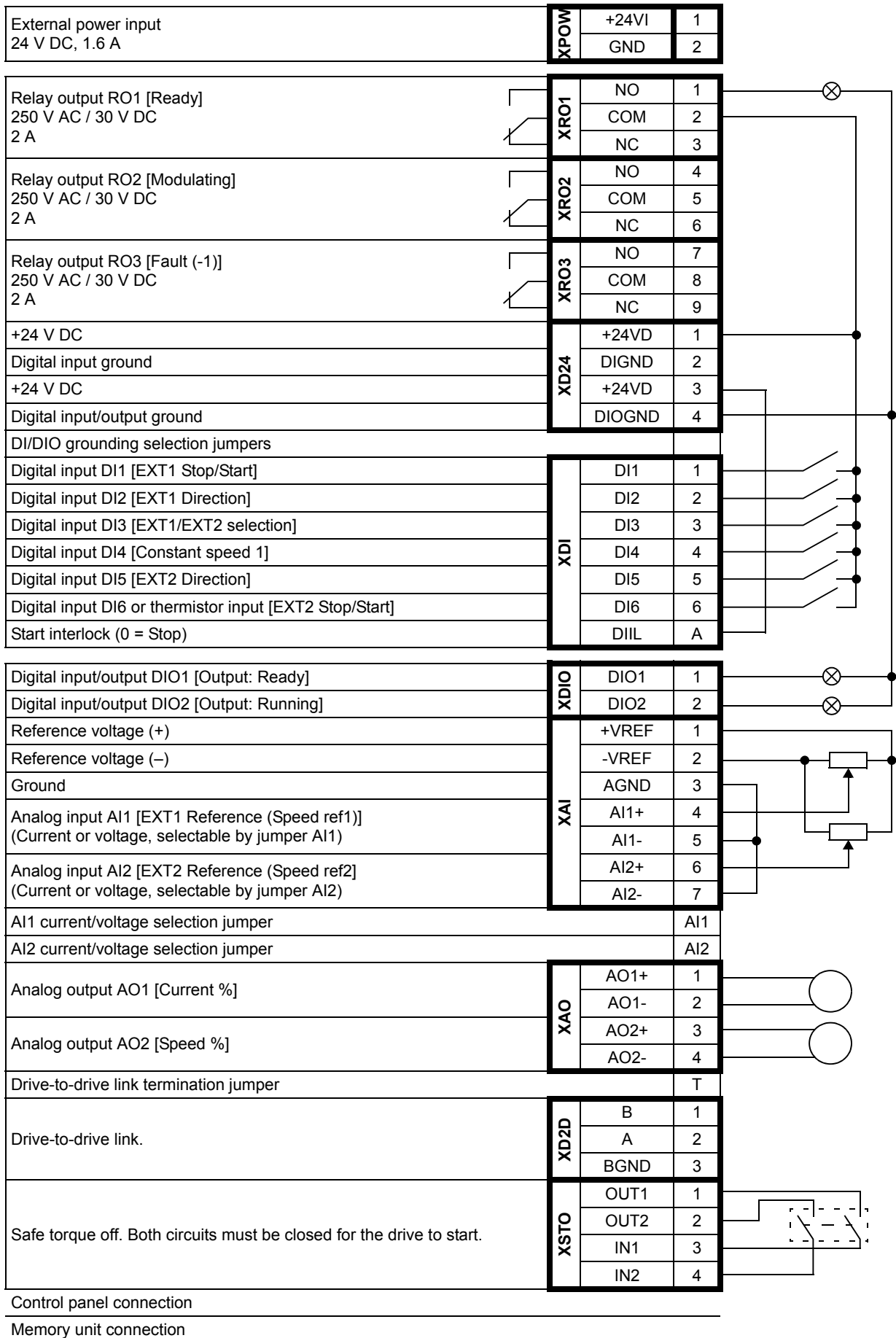
A constant speed (300 rpm) can be activated through DI4.

Default parameter settings for Hand/Auto macro

Below is a listing of default parameter values that differ from those listed in chapter [Additional parameter data](#) (page 257).

Parameter		Hand/Auto macro default
No.	Name	
10.01	<i>Ext1 start func</i>	<i>In1St In2Dir</i>
10.03	<i>Ext1 start in2</i>	<i>DI2</i>
10.04	<i>Ext2 start func</i>	<i>In1St In2Dir</i>
10.05	<i>Ext2 start in1</i>	<i>DI6</i>
10.06	<i>Ext2 start in2</i>	<i>DI5</i>
10.10	<i>Fault reset sel</i>	C.FALSE
12.01	<i>Ext1/Ext2 sel</i>	<i>DI3</i>
13.05	<i>AI1 min scale</i>	0.000
13.09	<i>AI2 max scale</i>	1500.000
13.10	<i>AI2 min scale</i>	0.000
21.02	<i>Speed ref2 sel</i>	<i>AI2 scaled</i>
21.04	<i>Speed ref1/2 sel</i>	<i>DI3</i>
26.02	<i>Const speed sel1</i>	<i>DI4</i>
26.06	<i>Const speed1</i>	300 rpm

Default control connections for the Hand/Auto macro



PID control macro

The PID control macro is suitable for process control applications, for example closed-loop pressure, level or flow control systems such as

- pressure boost pumps of municipal water supply systems
- level-controlling pumps of water reservoirs
- pressure boost pumps of district heating systems
- material flow control on a conveyor line.

The process reference signal is connected to analog input AI1 and the process feedback signal to AI2. Alternatively, a direct speed reference can be given to the drive through AI1. Then the PID controller is bypassed and the drive no longer controls the process variable.

Selection between direct speed control (control location EXT1) and process variable control (EXT2) is done through digital input DI3.

The stop/start signals for EXT1 and EXT2 are connected to DI1 and DI6 respectively.

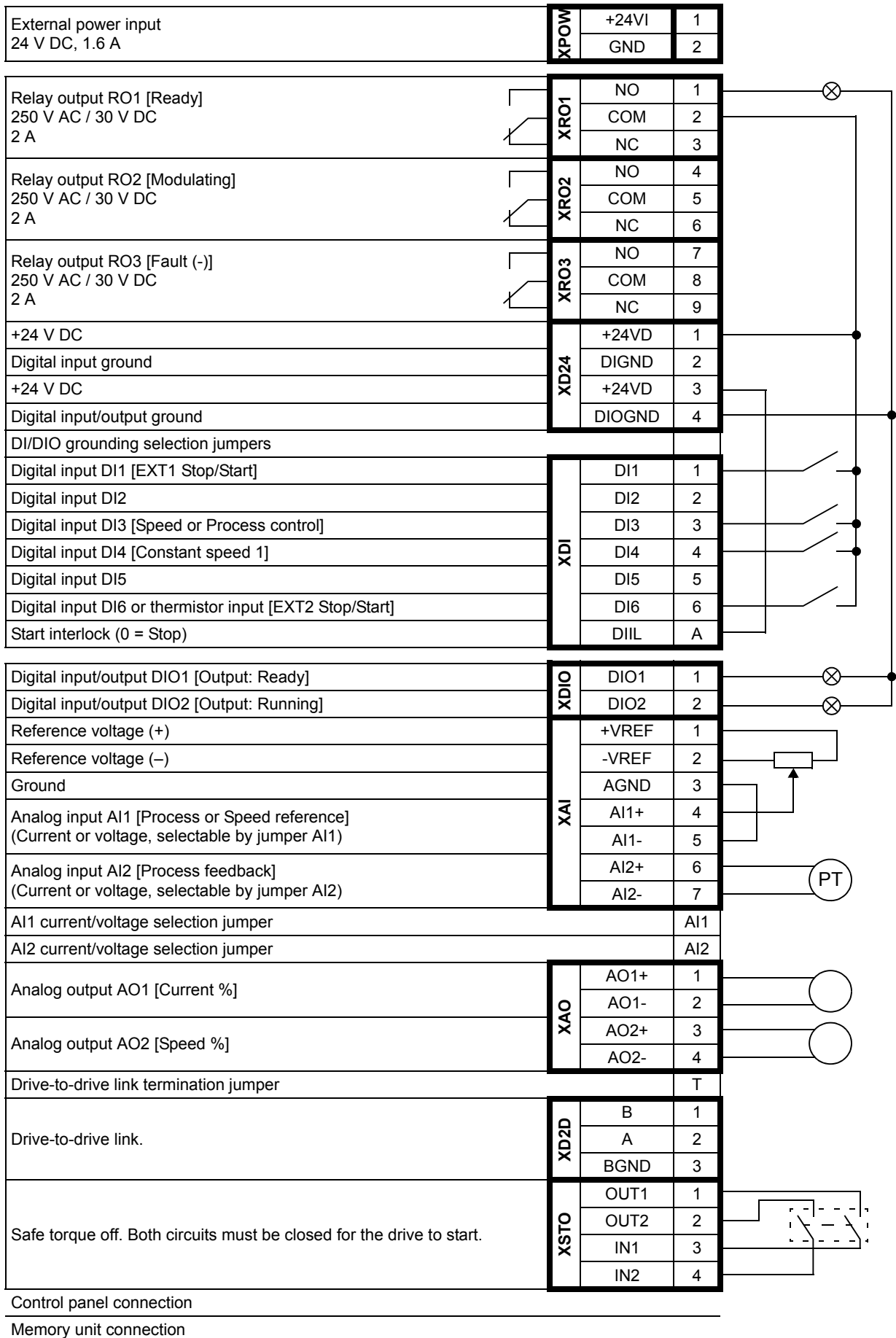
A constant speed (300 rpm) can be activated through DI4.

Default parameter settings for PID control macro

Below is a listing of default parameter values that differ from those listed in chapter [Additional parameter data](#) (page 257).

Parameter		PID control macro default
No.	Name	
10.04	<i>Ext2 start func</i>	<i>In1</i>
10.05	<i>Ext2 start in1</i>	<i>DI6</i>
10.10	<i>Fault reset sel</i>	C.FALSE
12.01	<i>Ext1/Ext2 sel</i>	<i>DI3</i>
13.05	<i>AI1 min scale</i>	0.000
13.09	<i>AI2 max scale</i>	1500.000
13.10	<i>AI2 min scale</i>	0.000
21.02	<i>Speed ref2 sel</i>	<i>PID out</i>
21.04	<i>Speed ref1/2 sel</i>	<i>DI3</i>
26.02	<i>Const speed sel1</i>	<i>DI4</i>
26.06	<i>Const speed1</i>	300 rpm

■ Default control connections for the PID control macro



Torque control macro

This macro is used in applications in which torque control of the motor is required. Torque reference is given through analog input AI2, typically as a current signal in the range of 0...20 mA (corresponding to 0...100% of rated motor torque).

The start/stop signal is connected to digital input DI1, direction signal to DI2. Through DI3, it is possible to select speed control instead of torque control.

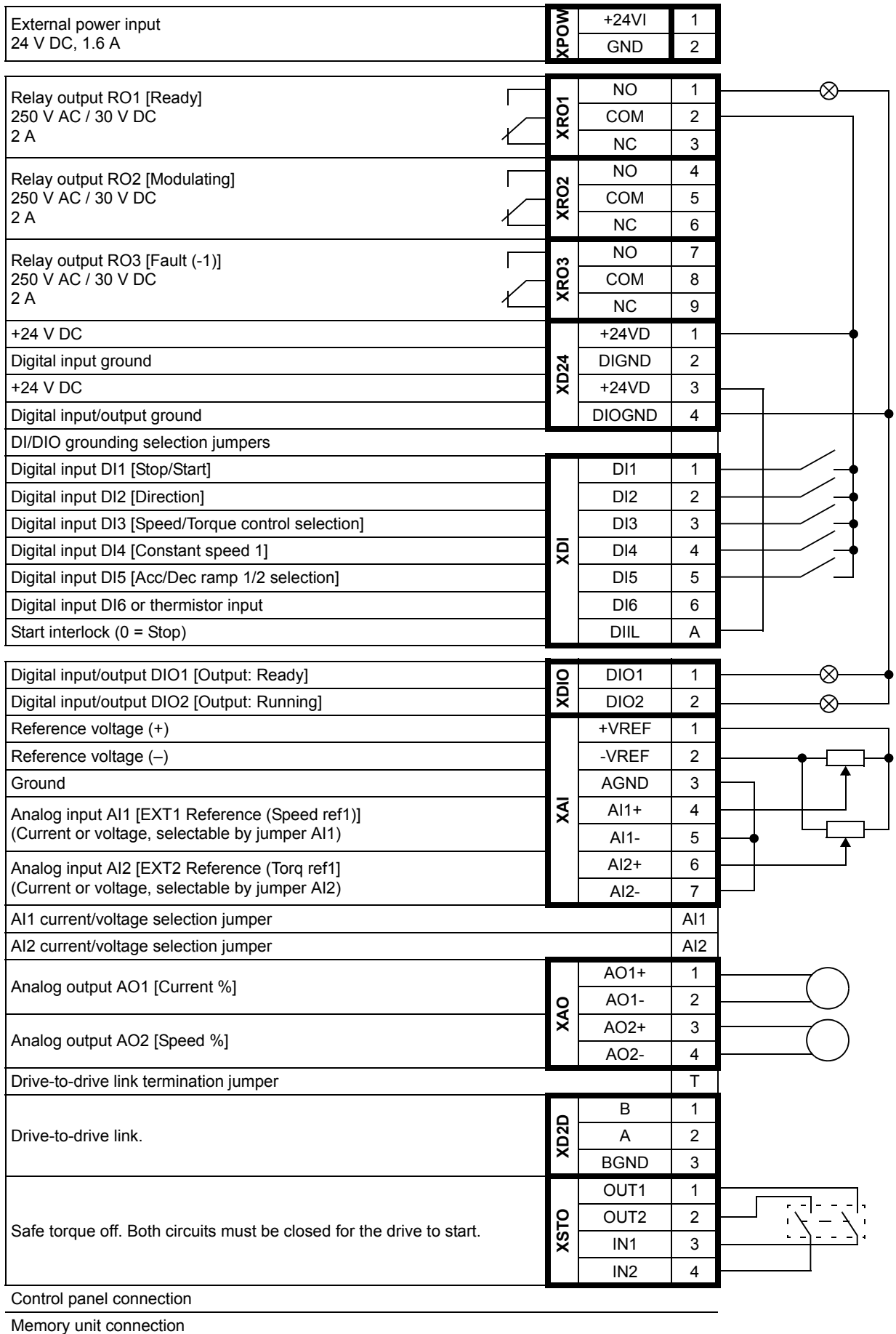
A constant speed (300 rpm) can be activated through DI4.

Default parameter settings for Torque control macro

Below is a listing of default parameter values that differ from those listed in chapter [Additional parameter data](#) (page 257).

Parameter		Torque control macro default
No.	Name	
10.01	<i>Ext1 start func</i>	<i>In1St In2Dir</i>
10.03	<i>Ext1 start in2</i>	<i>DI2</i>
10.04	<i>Ext2 start func</i>	<i>In1St In2Dir</i>
10.05	<i>Ext2 start in1</i>	<i>DI1</i>
10.06	<i>Ext2 start in2</i>	<i>DI2</i>
10.10	<i>Fault reset sel</i>	C.FALSE
12.01	<i>Ext1/Ext2 sel</i>	<i>DI3</i>
12.05	<i>Ext2 ctrl mode</i>	<i>Torque</i>
13.05	<i>AI1 min scale</i>	0.000
13.10	<i>AI2 min scale</i>	0.000
22.01	<i>Acc/Dec sel</i>	<i>DI5</i>
26.02	<i>Const speed sel1</i>	<i>DI4</i>
26.06	<i>Const speed1</i>	300 rpm

Default control connections for the Torque control macro



Sequential control macro

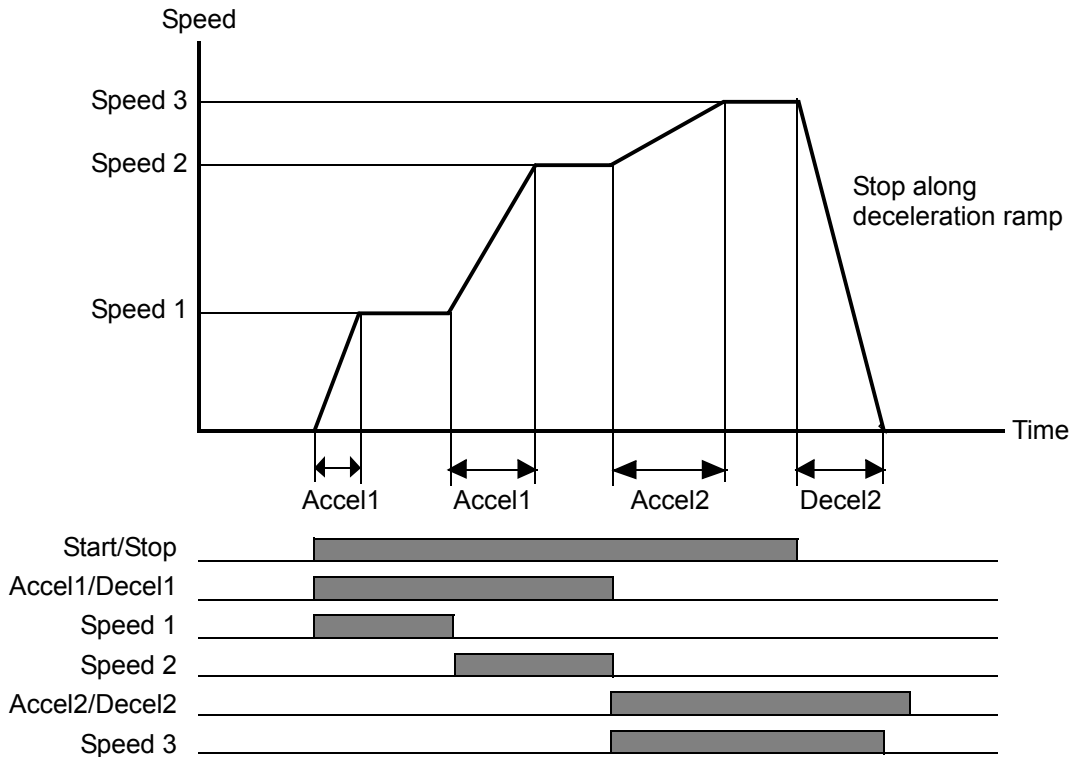
The Sequential control macro is suited for speed control applications in which speed reference, multiple constant speeds, and two acceleration and deceleration ramps can be used.

The macro offers seven preset constant speeds which can be activated by digital inputs DI4...DI6 (see parameter [26.01 Const speed func](#)). Two acceleration/ deceleration ramps are selectable through DI3.

An external speed reference can be given through analog input AI1. The reference is active only when no constant speed is activated (all of the digital inputs DI4...DI6 are off). Operational commands can also be given from the control panel.

Operation diagram

The figure below shows an example of the use of the macro.

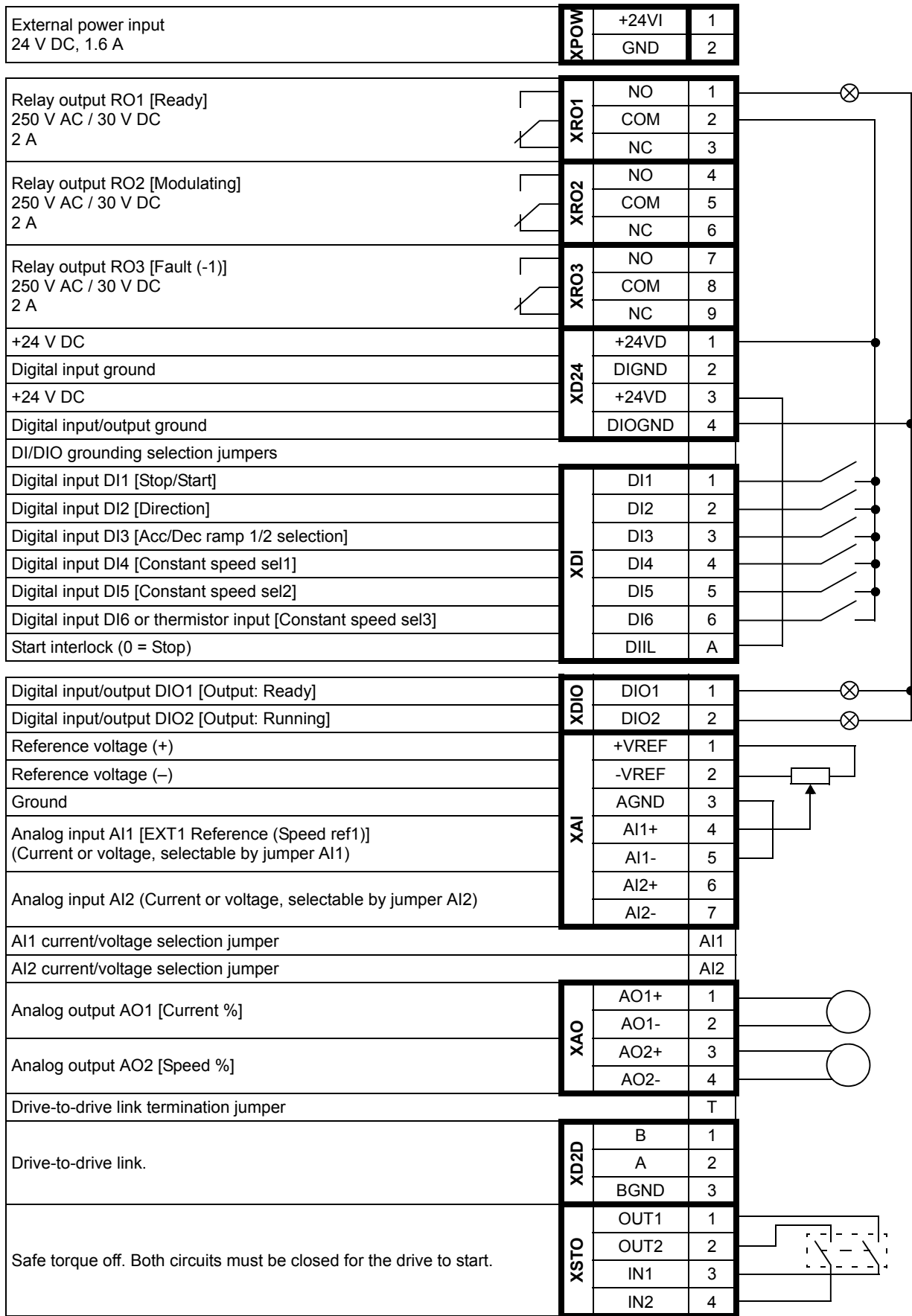


Default parameter settings for Sequential control macro

Below is a listing of default parameter values that differ from those listed in chapter [Additional parameter data](#) (page 257).

Parameter		Sequential control macro default
No.	Name	
10.01	<i>Ext1 start func</i>	<i>In1St In2Dir</i>
10.03	<i>Ext1 start in2</i>	<i>DI2</i>
10.10	<i>Fault reset sel</i>	C.FALSE
11.03	<i>Stop mode</i>	<i>Ramp</i>
13.05	<i>AI1 min scale</i>	0.000
22.01	<i>Acc/Dec sel</i>	<i>DI3</i>
26.01	<i>Const speed func</i>	0b01
26.02	<i>Const speed sel1</i>	<i>DI4</i>
26.03	<i>Const speed sel2</i>	<i>DI5</i>
26.04	<i>Const speed sel3</i>	<i>DI6</i>
26.06	<i>Const speed1</i>	300 rpm
26.07	<i>Const speed2</i>	600 rpm
26.08	<i>Const speed3</i>	900 rpm
26.09	<i>Const speed4</i>	1200 rpm
26.10	<i>Const speed5</i>	1500 rpm
26.11	<i>Const speed6</i>	2400 rpm
26.12	<i>Const speed7</i>	3000 rpm

Default control connections for the Sequential control macro



Control panel connection

Memory unit connection



Parameters [efesotomasyon.com](https://www.efesotomasyon.com)

What this chapter contains

The chapter describes the parameters, including actual signals, of the control program.

Note: By default, a selective list of parameters is shown by the drive panel or DriveStudio. All parameters can be displayed by setting parameter [16.15 Menu set sel](#) to [Load long](#).

Terms and abbreviations

Term	Definition
Actual signal	Type of parameter that is the result of a measurement or calculation by the drive. Actual signals can be monitored, but not adjusted, by the user. Parameter groups 1...9 typically contain actual signals.
Bit pointer setting	<p>A parameter setting that points to the value of a bit in another parameter (usually an actual signal), or that can be fixed to 0 (FALSE) or 1 (TRUE). When adjusting a bit pointer setting on the optional control panel, "Const" is selected in order to fix the value to 0 (displayed as "C.False") or 1 ("C.True"). "Pointer" is selected to define a source from another parameter.</p> <p>A pointer value is given in the format P.xx.yy.zz, where xx = parameter group, yy = parameter index, zz = bit number.</p> <p>Pointing to a nonexisting bit will be interpreted as 0 (FALSE).</p> <p>In addition to the "Const" and "Pointer" selections, bit pointer settings may also have other pre-selected settings.</p>
FbEq	Fieldbus equivalent. The scaling between the value shown on the panel and the integer used in serial communication.
p.u.	Per unit
Value pointer setting	<p>A parameter value that points to the value of another actual signal or parameter.</p> <p>A pointer value is given in the format P.xx.yy, where xx = parameter group, yy = parameter index.</p>

Parameter listing

No.	Name/Value	Description	FbEq
01 Actual values		Basic signals for monitoring of the drive.	
01.01	Motor speed rpm	Filtered actual speed in rpm. The used speed feedback is defined by parameter 19.02 Speed fb sel . The filter time constant can be adjusted using parameter 19.03 MotorSpeed filt .	100 = 1 rpm
01.02	Motor speed %	Actual speed in percent of the motor synchronous speed.	100 = 1%
01.03	Output frequency	Estimated drive output frequency in Hz.	100 = 1 Hz
01.04	Motor current	Measured motor current in A.	100 = 1 A
01.05	Motor current %	Motor current in percent of the nominal motor current.	10 = 1%
01.06	Motor torque	Motor torque in percent of the nominal motor torque. See also parameter 01.29 Torq nom scale .	10 = 1%
01.07	Dc-voltage	Measured intermediate circuit voltage.	100 = 1 V
01.08	Encoder1 speed	Encoder 1 speed in rpm.	100 = 1 rpm
01.09	Encoder1 pos	Actual position of encoder 1 within one revolution.	100000000 = 1 rev
01.10	Encoder2 speed	Encoder 2 speed in rpm.	100 = 1 rpm
01.11	Encoder2 pos	Actual position of encoder 2 within one revolution.	100000000 = 1 rev
01.12	Pos act	Actual position of encoder 1 in revolutions.	1000 = 1 rev
01.13	Pos 2nd enc	Scaled actual position of encoder 2 in revolutions.	1000 = 1 rev
01.14	Motor speed est	Estimated motor speed in rpm.	100 = 1 rpm
01.15	Temp inverter	Estimated IGBT temperature in percent of fault limit.	10 = 1%
01.16	Temp brk chopper	Brake chopper IGBT temperature in percent of fault limit.	10 = 1%
01.17	Motor temp1	Measured temperature of motor 1 in degrees Celsius when a KTY or Pt100 sensor is used. (With a PTC sensor, the value is always 0.)	10 = 1 °C
01.18	Motor temp2	Measured temperature of motor 2 in degrees Celsius when a KTY or Pt100 sensor is used. (With a PTC sensor, the value is always 0.)	10 = 1 °C
01.19	Used supply volt	Either the user-given supply voltage (parameter 47.04 Supply voltage), or, if auto-identification is enabled by parameter 47.03 SupplyVoltAutold , the automatically determined supply voltage.	10 = 1 V
01.20	Brake res load	Estimated temperature of the braking resistor. The value is given in percent of the temperature the resistor reaches when loaded with the power defined by parameter 48.04 Br power max cnt .	1 = 1%
01.21	Cpu usage	Microprocessor load in percent.	1 = 1%
01.22	Power inu out	Drive output power in kW or hp, depending on setting of parameter 16.17 Power unit .	100 = 1 kW or hp
01.23	Motor power	Measured motor shaft power in kW or hp, depending on setting of parameter 16.17 Power unit .	100 = 1 kW or hp
01.24	kWh inverter	Amount of energy that has passed through the drive (in either direction) in kilowatt-hours. The minimum value is zero. Can be reset by entering 0 using the DriveStudio PC tool.	1 = 1 kWh

104 Parameters

No.	Name/Value	Description	FbEq
01.25	kWh supply	Amount of energy that the drive has taken from (or given to) the AC supply in kilowatt-hours. Can be reset by entering 0 using the DriveStudio PC tool.	1 = 1 kWh
01.26	On-time counter	On-time counter. The counter runs when the drive is powered. Can be reset by entering 0 using the DriveStudio PC tool.	1 = 1 h
01.27	Run-time counter	Motor run-time counter. The counter runs when the inverter modulates. Can be reset by entering 0 using the DriveStudio PC tool.	1 = 1 h
01.28	Fan on-time	Running time of the drive cooling fan. Can be reset by entering 0 using the DriveStudio PC tool.	1 = 1 h
01.29	Torq nom scale	Nominal torque which corresponds to 100%. Note: This value is copied from parameter 99.12 Mot nom torque if entered. Otherwise the value is calculated.	1000 = 1 N•m
01.30	Polepairs	Calculated number of pole pairs in the motor.	1 = 1
01.31	Mech time const	Mechanical time constant of the drive and the machinery as determined by the speed controller autotune function. See parameter 23.20 PI tune mode .	1000 = 1 s
01.32	Temp phase A	Measured temperature of phase U power stage in percent of fault limit.	10 = 1%
01.33	Temp phase B	Measured temperature of phase V power stage in percent of fault limit.	10 = 1%
01.34	Temp phase C	Measured temperature of phase W power stage in percent below fault limit.	10 = 1%
01.35	Saved energy	Energy saved in kWh compared to direct-on-line motor connection. See parameter group 45 Energy optimising on page 227 .	1 = 1 kWh
01.36	Saved amount	Monetary savings compared to direct-on-line motor connection. This value is a multiplication of parameters 01.35 Saved energy and 45.02 Energy tariff1 . See parameter group 45 Energy optimising on page 227 .	1 = 1
01.37	Saved CO2	Reduction in CO ₂ emissions in metric tons compared to direct-on-line motor connection. This value is calculated by multiplying the saved energy in MWh by 45.07 CO2 Conv factor (default 0.5 tn/MWh). See parameter group 45 Energy optimising on page 227 .	1 = 1 metric ton
01.38	Temp int board	Measured temperature of the interface board in degrees Celsius.	10 = 1 °C
02 I/O values		Input and output signals.	
02.01	DI status	Status of digital inputs DI8...DI1. The 7th digit reflects the start interlock input (DIIL). Example: 01000001 = DI1 and DIIL are on, DI2...DI6 and DI8 are off.	-
02.02	RO status	Status of relay outputs RO7...RO1. Example: 0000001 = RO1 is energized, RO2...RO7 are de-energized.	-
02.03	DIO status	Status of digital input/outputs DIO10...DIO1. Example: 0000001001 = DIO1 and DIO4 are on, remainder are off. DIO3...DIO10 are available only with FIO I/O extension modules.	-
02.04	AI1	Value of analogue input AI1 in V or mA. Input type is selected with jumper J1 on the JCU Control Unit.	1000 = 1 unit

No.	Name/Value	Description	FbEq
02.05	AI1 scaled	Scaled value of analogue input AI1. See parameters 13.04 AI1 max scale and 13.05 AI1 min scale .	1000 = 1 unit
02.06	AI2	Value of analogue input AI2 in V or mA. Input type is selected with jumper J2 on the JCU Control Unit.	1000 = 1 unit
02.07	AI2 scaled	Scaled value of analogue input AI2. See parameters 13.09 AI2 max scale and 13.10 AI2 min scale .	1000 = 1 unit
02.08	AI3	Value of analogue input AI3 in V or mA. For input type information, see the extension module manual.	1000 = 1 unit
02.09	AI3 scaled	Scaled value of analogue input AI3. See parameters 13.14 AI3 max scale and 13.15 AI3 min scale .	1000 = 1 unit
02.10	AI4	Value of analogue input AI4 in V or mA. For input type information, see the extension module manual.	1000 = 1 unit
02.11	AI4 scaled	Scaled value of analogue input AI4. See parameters 13.19 AI4 max scale and 13.20 AI4 min scale .	1000 = 1 unit
02.12	AI5	Value of analogue input AI5 in V or mA. For input type information, see the extension module manual.	1000 = 1 unit
02.13	AI5 scaled	Scaled value of analogue input AI5. See parameters 13.24 AI5 max scale and 13.25 AI5 min scale .	1000 = 1 unit
02.14	AI6	Value of analogue input AI6 in V or mA. For input type information, see the extension module manual.	1000 = 1 unit
02.15	AI6 scaled	Scaled value of analogue input AI6. See parameters 13.29 AI6 max scale and 13.30 AI6 min scale .	1000 = 1 unit
02.16	AO1	Value of analogue output AO1 in mA.	1000 = 1 mA
02.17	AO2	Value of analogue output AO2 in mA.	1000 = 1 mA
02.18	AO3	Value of analogue output AO3 in mA.	1000 = 1 mA
02.19	AO4	Value of analogue output AO4 in mA.	1000 = 1 mA
02.20	Freq in	Scaled value of DIO1 when it is used as a frequency input. See parameters 14.02 DIO1 conf and 14.57 Freq in max .	1000 = 1
02.21	Freq out	Frequency output value of DIO2 when it is used as a frequency output (parameter 14.06 is set to <i>Freq output</i>).	1000 = 1 Hz

No.	Name/Value	Description	FbEq		
02.22	FBA main cw	Internal Control Word of the drive received through the fieldbus adapter interface. See also chapter Control through a fieldbus adapter on page 331. Log. = Logical combination (i.e. Bit AND/OR Selection parameter); Par. = Selection parameter.	-		
Bit	Name	Value	Information	Log.	Par.
0*	Stop	1	Stop according to the stop mode selected by par. 11.03 Stop mode or according to the requested stop mode (bits 2...6). Note: Simultaneous stop and start commands result in a stop command.	OR	10.01 , 10.04
		0	No action.		
1	Start	1	Start. Note: Simultaneous stop and start commands result in a stop command.	OR	10.01 , 10.04
		0	No action.		
2*	StpMode em off	1	Emergency OFF2 (bit 0 must be 1). Drive is stopped by cutting off motor power supply (the motor coasts to stop). The drive will restart only with the next rising edge of the start signal when the run enable signal is on.	AND	-
		0	No action.		
3*	StpMode em stop	1	Emergency stop OFF3 (bit 0 must be 1). Stop within time defined by 22.12 Em stop time .	AND	10.13
		0	No action.		
4*	StpMode off1	1	Emergency stop OFF1 (bit 0 must be 1). Stop along the currently active deceleration ramp.	AND	10.15
		0	No action.		
5*	StpMode ramp	1	Stop along the currently active deceleration ramp.	-	11.03
		0	No action.		
6*	StpMode coast	1	Coast to stop.	-	11.03
		0	No action.		
7	Run enable	1	Activate run enable.	AND	10.11
		0	Activate run disable.		
8	Reset	0 -> 1	Fault reset if an active fault exists.	OR	10.10
		other	No action.		
(continued)					
* If all stop mode bits (2...6) are 0, stop mode is selected by parameter 11.03 Stop mode . Coast stop (bit 6) overrides the emergency stop (bits 2/3/4). Emergency stop overrides normal ramp stop (bit 5).					

No.	Name/Value	Description		FbEq	
Bit	Name	Value	Information	Log.	Par.
(continued)					
9	Jogging 1	1	Activate Jogging 1. See section Jogging on page 64.	OR	10.07
		0	Jogging 1 disabled.		
10	Jogging 2	1	Activate Jogging 2. See section Jogging on page 64.	OR	10.08
		0	Jogging 2 disabled.		
11	Remote cmd	1	Fieldbus control enabled.	-	-
		0	Fieldbus control disabled.		
12	Ramp out 0	1	Force output of Ramp Function Generator to zero. The drive ramps to a stop (current and DC voltage limits are in force).	-	-
		0	No action.		
13	Ramp hold	1	Halt ramping (Ramp Function Generator output held).	-	-
		0	No action.		
14	Ramp in 0	1	Force input of Ramp Function Generator to zero.	-	-
		0	No action.		
15	Ext1 / Ext2	1	Switch to external control location EXT2.	OR	12.01
		0	Switch to external control location EXT1.		
16	Req startinh	1	Activate start inhibit.	-	-
		0	No start inhibit.		
17	Local ctl	1	Request local control for Control Word. Used when the drive is controlled from a PC tool or panel or local fieldbus. <ul style="list-style-type: none"> Local fieldbus: Transfer to fieldbus local control (control through Control Word or reference). Fieldbus steals the control. Panel or PC tool: Transfer to local control. 	-	-
		0	Request external control.		
18	FbLocal ref	1	Request fieldbus local control.	-	-
		0	No fieldbus local control.		
19...27	Reserved				
28	CW B28	Freely programmable control bits. See parameters 50.08...50.11 and the user manual of the fieldbus adapter.		-	-
29	CW B29				
30	CW B30				
31	CW B31				

No.	Name/Value	Description	FbEq																																																										
02.24	FBA main sw	Internal Status word of the drive to be sent through the fieldbus adapter interface. See also chapter Control through a fieldbus adapter on page 331.	-																																																										
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Name</th> <th>Value</th> <th>Information</th> </tr> </thead> <tbody> <tr> <td rowspan="2">0</td> <td rowspan="2">Ready</td> <td>1</td> <td>Drive is ready to receive start command.</td> </tr> <tr> <td>0</td> <td>Drive is not ready.</td> </tr> <tr> <td rowspan="2">1</td> <td rowspan="2">Enabled</td> <td>1</td> <td>External run enable signal is received.</td> </tr> <tr> <td>0</td> <td>No external run enable signal is received.</td> </tr> <tr> <td rowspan="2">2</td> <td rowspan="2">Running</td> <td>1</td> <td>Drive is modulating.</td> </tr> <tr> <td>0</td> <td>Drive is not modulating.</td> </tr> <tr> <td rowspan="2">3</td> <td rowspan="2">Ref running</td> <td>1</td> <td>Normal operation is enabled. Drive is running and following given reference.</td> </tr> <tr> <td>0</td> <td>Normal operation is disabled. Drive is not following given reference (for example, it is modulating during magnetization).</td> </tr> <tr> <td rowspan="2">4</td> <td rowspan="2">Em off (OFF2)</td> <td>1</td> <td>Emergency OFF2 is active.</td> </tr> <tr> <td>0</td> <td>Emergency OFF2 is inactive.</td> </tr> <tr> <td rowspan="2">5</td> <td rowspan="2">Em stop (OFF3)</td> <td>1</td> <td>Emergency stop OFF3 (ramp stop) is active.</td> </tr> <tr> <td>0</td> <td>Emergency stop OFF3 is inactive.</td> </tr> <tr> <td rowspan="2">6</td> <td rowspan="2">Ack startinh</td> <td>1</td> <td>Start inhibit is active.</td> </tr> <tr> <td>0</td> <td>Start inhibit is inactive.</td> </tr> <tr> <td rowspan="2">7</td> <td rowspan="2">Alarm</td> <td>1</td> <td>An alarm is active. See chapter Fault tracing on page 283</td> </tr> <tr> <td>0</td> <td>No alarm is active.</td> </tr> <tr> <td rowspan="2">8</td> <td rowspan="2">At setpoint</td> <td>1</td> <td>Drive is at setpoint. Actual value equals reference value (i.e. the difference between the actual speed and speed reference is within the speed window defined by parameter 19.10 Speed window).</td> </tr> <tr> <td>0</td> <td>Drive has not reached setpoint.</td> </tr> </tbody> </table>	Bit	Name	Value	Information	0	Ready	1	Drive is ready to receive start command.	0	Drive is not ready.	1	Enabled	1	External run enable signal is received.	0	No external run enable signal is received.	2	Running	1	Drive is modulating.	0	Drive is not modulating.	3	Ref running	1	Normal operation is enabled. Drive is running and following given reference.	0	Normal operation is disabled. Drive is not following given reference (for example, it is modulating during magnetization).	4	Em off (OFF2)	1	Emergency OFF2 is active.	0	Emergency OFF2 is inactive.	5	Em stop (OFF3)	1	Emergency stop OFF3 (ramp stop) is active.	0	Emergency stop OFF3 is inactive.	6	Ack startinh	1	Start inhibit is active.	0	Start inhibit is inactive.	7	Alarm	1	An alarm is active. See chapter Fault tracing on page 283	0	No alarm is active.	8	At setpoint	1	Drive is at setpoint. Actual value equals reference value (i.e. the difference between the actual speed and speed reference is within the speed window defined by parameter 19.10 Speed window).	0	Drive has not reached setpoint.	
Bit	Name	Value	Information																																																										
0	Ready	1	Drive is ready to receive start command.																																																										
		0	Drive is not ready.																																																										
1	Enabled	1	External run enable signal is received.																																																										
		0	No external run enable signal is received.																																																										
2	Running	1	Drive is modulating.																																																										
		0	Drive is not modulating.																																																										
3	Ref running	1	Normal operation is enabled. Drive is running and following given reference.																																																										
		0	Normal operation is disabled. Drive is not following given reference (for example, it is modulating during magnetization).																																																										
4	Em off (OFF2)	1	Emergency OFF2 is active.																																																										
		0	Emergency OFF2 is inactive.																																																										
5	Em stop (OFF3)	1	Emergency stop OFF3 (ramp stop) is active.																																																										
		0	Emergency stop OFF3 is inactive.																																																										
6	Ack startinh	1	Start inhibit is active.																																																										
		0	Start inhibit is inactive.																																																										
7	Alarm	1	An alarm is active. See chapter Fault tracing on page 283																																																										
		0	No alarm is active.																																																										
8	At setpoint	1	Drive is at setpoint. Actual value equals reference value (i.e. the difference between the actual speed and speed reference is within the speed window defined by parameter 19.10 Speed window).																																																										
		0	Drive has not reached setpoint.																																																										
(continued)																																																													

No.	Name/Value	Description	FbEq
	Bit	Name	Value Information
	(continued)		
9	Limit	1	Operation is limited by any of the torque limits.
		0	Operation is within the torque limits.
10	Above limit	1	Actual speed exceeds limit defined by parameter 19.08 Above speed lim.
		0	Actual speed is within the defined limits.
11	Ext2 act	1	External control location EXT2 is active.
		0	External control location EXT1 is active.
12	Local fb	1	Fieldbus local control is active.
		0	Fieldbus local control is inactive.
13	Zero speed	1	Drive speed is below limit defined by parameter 19.06 Zero speed limit.
		0	Drive has not reached zero speed limit.
14	Rev act	1	Drive is running in reverse direction.
		0	Drive is running in forward direction.
15	Reserved		
16	Fault	1	A fault is active. See chapter Fault tracing on page 283 .
		0	No fault is active.
17	Local panel	1	Local control is active, i.e. the drive is controlled from PC tool or control panel.
		0	Local control is inactive.
18...26	Reserved		
27	Request ctrl	1	Control Word is requested from fieldbus.
		0	Control Word is not requested from fieldbus.
28	SW B28	Programmable control bits (unless fixed by the used profile). See parameters 50.08...50.11 and the user manual of the fieldbus adapter.	
29	SW B29		
30	SW B30		
31	SW B31		
02.26	FBA main ref1	Internal and scaled reference 1 of the drive received through the fieldbus adapter interface. See parameter 50.04 Fb ref1 modesel and chapter Control through a fieldbus adapter on page 331 .	1 = 1
02.27	FBA main ref2	Internal and scaled reference 2 of the drive received through the fieldbus adapter interface. See parameter 50.05 Fb ref2 modesel and chapter Control through a fieldbus adapter on page 331 .	1 = 1
02.30	D2D main cw	Drive-to-drive control word received from the master. See also actual signal 02.31 D2D follower cw .	-
	Bit	Information	
	0	Stop.	
	1	Start.	
	2 ... 6	Reserved.	
	7	Run enable. By default, not connected in a follower drive.	
	8	Reset. By default, not connected in a follower drive.	
	9 ... 14	Freely assignable through bit pointer settings.	
	15	EXT1/EXT2 selection. 0 = EXT1 active, 1 = EXT2 active. By default, not connected in a follower drive.	

110 Parameters

No.	Name/Value	Description	FbEq																
02.31	D2D follower cw	Drive-to-drive control word sent to the followers by default. See also parameter group 57 D2D communication on page 235 .	-																
<table border="1"> <thead> <tr> <th>Bit</th> <th>Information</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Stop.</td> </tr> <tr> <td>1</td> <td>Start.</td> </tr> <tr> <td>2 ... 6</td> <td>Reserved.</td> </tr> <tr> <td>7</td> <td>Run enable.</td> </tr> <tr> <td>8</td> <td>Reset.</td> </tr> <tr> <td>9 ... 14</td> <td>Freely assignable through bit pointer settings.</td> </tr> <tr> <td>15</td> <td>EXT1/EXT2 selection. 0 = EXT1 active, 1 = EXT2 active.</td> </tr> </tbody> </table>				Bit	Information	0	Stop.	1	Start.	2 ... 6	Reserved.	7	Run enable.	8	Reset.	9 ... 14	Freely assignable through bit pointer settings.	15	EXT1/EXT2 selection. 0 = EXT1 active, 1 = EXT2 active.
Bit	Information																		
0	Stop.																		
1	Start.																		
2 ... 6	Reserved.																		
7	Run enable.																		
8	Reset.																		
9 ... 14	Freely assignable through bit pointer settings.																		
15	EXT1/EXT2 selection. 0 = EXT1 active, 1 = EXT2 active.																		
02.32	D2D ref1	Drive-to-drive reference 1 received from the master.	1 = 1																
02.33	D2D ref2	Drive-to-drive reference 2 received from the master.	1 = 1																
02.34	Panel ref	Reference given from the control panel. See also parameter 56.07 Local ref unit .	100 = 1 rpm 10 = 1%																
02.35	FEN DI status	Status of the digital inputs of FEN-xx encoder interfaces in drive option slots 1 and 2. Examples: 000001 (01h) = DI1 of FEN-xx in slot 1 is ON, all others are OFF. 000010 (02h) = DI2 of FEN-xx in slot 1 is ON, all others are OFF. 010000 (10h) = DI1 of FEN-xx in slot 2 is ON, all others are OFF. 100000 (20h) = DI2 of FEN-xx in slot 2 is on, all others are OFF.	-																

No.	Name/Value	Description	FbEq		
02.36	EFB main cw	Internal Control Word of the drive received through the embedded fieldbus interface. See chapter Control through the embedded fieldbus interface on page 303. Log. = Logical combination (i.e. Bit AND/OR Selection parameter); Par. = Selection parameter.	-		
Bit	Name	Value	Information	Log.	Par.
0*	Stop	1	Stop according to the stop mode selected by par. 11.03 Stop mode or according to the requested stop mode (bits 2...6). Note: Simultaneous stop and start commands result in a stop command.	OR	10.01 , 10.04
		0	No action.		
1	Start	1	Start. Note: Simultaneous stop and start commands result in a stop command.	OR	10.01 , 10.04
		0	No action.		
2*	StpMode em off	1	Emergency OFF2 (bit 0 must be 1). Drive is stopped by cutting off motor power supply (the motor coasts to stop). The drive will restart only with the next rising edge of the start signal when the run enable signal is on.	AND	-
		0	No action.		
3*	StpMode em stop	1	Emergency stop OFF3 (bit 0 must be 1). Stop within time defined by 22.12 Em stop time .	AND	10.13
		0	No action.		
4*	StpMode off1	1	Emergency stop OFF1 (bit 0 must be 1). Stop along the currently active deceleration ramp.	AND	10.15
		0	No action.		
5*	StpMode ramp	1	Stop along the currently active deceleration ramp.	-	11.03
		0	No action.		
6*	StpMode coast	1	Coast to stop.	-	11.03
		0	No action.		
7	Run enable	1	Activate run enable.	AND	10.11
		0	Activate run disable.		
8	Reset	0 -> 1	Fault reset if an active fault exists.	OR	10.10
		other	No action.		
(continued)					
* If all stop mode bits (2...6) are 0, stop mode is selected by parameter 11.03 Stop mode . Coast stop (bit 6) overrides the emergency stop (bits 2/3/4). Emergency stop overrides normal ramp stop (bit 5).					

112 Parameters

No.	Name/Value	Description	FbEq			
	Bit	Name	Value	Information	Log.	Par.
	(continued)					
9	Jogging 1	1	Activate Jogging 1. See section Jogging on page 64.	OR	10.07	
		0	Jogging 1 disabled.			
10	Jogging 2	1	Activate Jogging 2. See section Jogging on page 64.	OR	10.08	
		0	Jogging 2 disabled.			
11	Remote cmd	1	Fieldbus control enabled.	-	-	
		0	Fieldbus control disabled.			
12	Ramp out 0	1	Force output of Ramp Function Generator to zero. The drive ramps to a stop (current and DC voltage limits are in force).	-	-	
		0	No action.			
13	Ramp hold	1	Halt ramping (Ramp Function Generator output held).	-	-	
		0	No action.			
14	Ramp in 0	1	Force input of Ramp Function Generator to zero.	-	-	
		0	No action.			
15	Ext1 / Ext2	1	Switch to external control location EXT2.	OR	12.01	
		0	Switch to external control location EXT1.			
16	Req startinh	1	Activate start inhibit.	-	-	
		0	No start inhibit.			
17	Local ctl	1	Request local control for Control Word. Used when the drive is controlled from a PC tool or panel or local fieldbus. <ul style="list-style-type: none"> Local fieldbus: Transfer to fieldbus local control (control through Control Word or reference). Fieldbus steals the control. Panel or PC tool: Transfer to local control. 	-	-	
		0	Request external control.			
18	FbLocal ref	1	Request fieldbus local control.	-	-	
		0	No fieldbus local control.			
19...27	Reserved					
28	CW B28	Freely programmable control bits. See parameters 50.08...50.11 .			-	-
29	CW B29					
30	CW B30					
31	CW B31					

No.	Name/Value	Description	FbEq																																																										
02.37	EFB main sw	Internal Status word of the drive to be sent through the embedded fieldbus interface. See chapter Control through the embedded fieldbus interface on page 303.	-																																																										
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Name</th> <th>Value</th> <th>Information</th> </tr> </thead> <tbody> <tr> <td rowspan="2">0</td> <td rowspan="2">Ready</td> <td>1</td> <td>Drive is ready to receive start command.</td> </tr> <tr> <td>0</td> <td>Drive is not ready.</td> </tr> <tr> <td rowspan="2">1</td> <td rowspan="2">Enabled</td> <td>1</td> <td>External run enable signal is received.</td> </tr> <tr> <td>0</td> <td>No external run enable signal is received.</td> </tr> <tr> <td rowspan="2">2</td> <td rowspan="2">Running</td> <td>1</td> <td>Drive is modulating.</td> </tr> <tr> <td>0</td> <td>Drive is not modulating.</td> </tr> <tr> <td rowspan="2">3</td> <td rowspan="2">Ref running</td> <td>1</td> <td>Normal operation is enabled. Drive is running and following given reference.</td> </tr> <tr> <td>0</td> <td>Normal operation is disabled. Drive is not following given reference (for example, it is modulating during magnetization).</td> </tr> <tr> <td rowspan="2">4</td> <td rowspan="2">Em off (OFF2)</td> <td>1</td> <td>Emergency OFF2 is active.</td> </tr> <tr> <td>0</td> <td>Emergency OFF2 is inactive.</td> </tr> <tr> <td rowspan="2">5</td> <td rowspan="2">Em stop (OFF3)</td> <td>1</td> <td>Emergency stop OFF3 (ramp stop) is active.</td> </tr> <tr> <td>0</td> <td>Emergency stop OFF3 is inactive.</td> </tr> <tr> <td rowspan="2">6</td> <td rowspan="2">Ack startinh</td> <td>1</td> <td>Start inhibit is active.</td> </tr> <tr> <td>0</td> <td>Start inhibit is inactive.</td> </tr> <tr> <td rowspan="2">7</td> <td rowspan="2">Alarm</td> <td>1</td> <td>An alarm is active. See chapter Fault tracing on page 283.</td> </tr> <tr> <td>0</td> <td>No alarm is active.</td> </tr> <tr> <td rowspan="2">8</td> <td rowspan="2">At setpoint</td> <td>1</td> <td>Drive is at setpoint. Actual value equals reference value (i.e. the difference between the actual speed and speed reference is within the speed window defined by parameter 19.10 Speed window).</td> </tr> <tr> <td>0</td> <td>Drive has not reached setpoint.</td> </tr> </tbody> </table>	Bit	Name	Value	Information	0	Ready	1	Drive is ready to receive start command.	0	Drive is not ready.	1	Enabled	1	External run enable signal is received.	0	No external run enable signal is received.	2	Running	1	Drive is modulating.	0	Drive is not modulating.	3	Ref running	1	Normal operation is enabled. Drive is running and following given reference.	0	Normal operation is disabled. Drive is not following given reference (for example, it is modulating during magnetization).	4	Em off (OFF2)	1	Emergency OFF2 is active.	0	Emergency OFF2 is inactive.	5	Em stop (OFF3)	1	Emergency stop OFF3 (ramp stop) is active.	0	Emergency stop OFF3 is inactive.	6	Ack startinh	1	Start inhibit is active.	0	Start inhibit is inactive.	7	Alarm	1	An alarm is active. See chapter Fault tracing on page 283.	0	No alarm is active.	8	At setpoint	1	Drive is at setpoint. Actual value equals reference value (i.e. the difference between the actual speed and speed reference is within the speed window defined by parameter 19.10 Speed window).	0	Drive has not reached setpoint.	
Bit	Name	Value	Information																																																										
0	Ready	1	Drive is ready to receive start command.																																																										
		0	Drive is not ready.																																																										
1	Enabled	1	External run enable signal is received.																																																										
		0	No external run enable signal is received.																																																										
2	Running	1	Drive is modulating.																																																										
		0	Drive is not modulating.																																																										
3	Ref running	1	Normal operation is enabled. Drive is running and following given reference.																																																										
		0	Normal operation is disabled. Drive is not following given reference (for example, it is modulating during magnetization).																																																										
4	Em off (OFF2)	1	Emergency OFF2 is active.																																																										
		0	Emergency OFF2 is inactive.																																																										
5	Em stop (OFF3)	1	Emergency stop OFF3 (ramp stop) is active.																																																										
		0	Emergency stop OFF3 is inactive.																																																										
6	Ack startinh	1	Start inhibit is active.																																																										
		0	Start inhibit is inactive.																																																										
7	Alarm	1	An alarm is active. See chapter Fault tracing on page 283.																																																										
		0	No alarm is active.																																																										
8	At setpoint	1	Drive is at setpoint. Actual value equals reference value (i.e. the difference between the actual speed and speed reference is within the speed window defined by parameter 19.10 Speed window).																																																										
		0	Drive has not reached setpoint.																																																										
(continued)																																																													

No.	Name/Value	Description	FbEq
	Bit	Name	Value Information
	(continued)		
9	Limit	1	Operation is limited by any of the torque limits.
		0	Operation is within the torque limits.
10	Above limit	1	Actual speed exceeds limit defined by parameter 19.08 Above speed lim.
		0	Actual speed is within the defined limits.
11	Ext2 act	1	External control location EXT2 is active.
		0	External control location EXT1 is active.
12	Local fb	1	Fieldbus local control is active.
		0	Fieldbus local control is inactive.
13	Zero speed	1	Drive speed is below limit defined by parameter 19.06 Zero speed limit.
		0	Drive has not reached zero speed limit.
14	Rev act	1	Drive is running in reverse direction.
		0	Drive is running in forward direction.
15	Reserved		
16	Fault	1	A fault is active. See chapter Fault tracing on page 283 .
		0	No fault is active.
17	Local panel	1	Local control is active, i.e. the drive is controlled from PC tool or control panel.
		0	Local control is inactive.
18...26	Reserved		
27	Request ctl	1	Control Word is requested from fieldbus.
		0	Control Word is not requested from fieldbus.
28	SW B28	Programmable control bits (unless fixed by the used profile). See parameters 50.08...50.11 .	
29	SW B29		
30	SW B30		
31	SW B31		
02.38	EFB main ref1	Internal and scaled reference 1 of the drive received through the embedded fieldbus interface. See parameter 50.04 Fb ref1 modesel and chapter Control through the embedded fieldbus interface on page 303 .	-
02.39	EFB main ref2	Internal and scaled reference 2 of the drive received through the embedded fieldbus interface. See parameter 50.05 Fb ref2 modesel and chapter Control through the embedded fieldbus interface on page 303 .	--
03 Control values		Speed control, torque control, and other values.	
03.03	SpeedRef unramp	Used speed reference before ramping and shaping in rpm.	100 = 1 rpm
03.05	SpeedRef ramped	Ramped and shaped speed reference in rpm.	100 = 1 rpm
03.06	SpeedRef used	Used speed reference in rpm (reference before speed error calculation).	100 = 1 rpm
03.07	Speed error filt	Filtered speed error value in rpm.	100 = 1 rpm
03.08	Acc comp torq	Output of the acceleration compensation (torque in percent).	10 = 1%
03.09	Torq ref sp ctrl	Limited speed controller output torque in percent.	10 = 1%
03.11	Torq ref ramped	Ramped torque reference in percent.	10 = 1%

No.	Name/Value	Description	FbEq
03.12	Torq ref sp lim	Torque reference limited by the rush control (value in percent). Torque is limited to ensure that the speed is between the minimum and maximum speed limits defined by parameters 20.01 Maximum speed and 20.02 Minimum speed .	10 = 1%
03.13	Torq ref to TC	Torque reference in percent for the torque control.	10 = 1%
03.14	Torq ref used	Torque reference after frequency, voltage and torque limiters. 100% corresponds to the motor nominal torque.	10 = 1%
03.15	Brake torq mem	Torque value (in percent) stored when the mechanical brake close command is issued.	10 = 1%
03.16	Brake command	Brake on/off command; 0 = close, 1 = open. For brake on/off control, connect this signal to a relay output (or digital output). See section Mechanical brake control on page 72.	1 = 1
03.17	Flux actual	Actual flux reference in percent.	1 = 1%
03.18	Speed ref pot	Output of the motor potentiometer function. (The motor potentiometer is configured using parameters 21.10...21.12 .)	100 = 1 rpm
03.20	Max speed ref	Maximum speed reference.	100 = 1 rpm
03.21	Min speed ref	Minimum speed reference.	100 = 1 rpm
04 Appl values		Process and counter values.	
04.01	Process act1	Process feedback 1 for the process PID controller.	100 = 1 unit
04.02	Process act2	Process feedback 2 for the process PID controller.	100 = 1 unit
04.03	Process act	Final process feedback after process feedback selection and modification.	100 = 1 unit
04.04	Process PID err	Process PID error, i.e. difference between PID setpoint and feedback.	10 = 1 unit
04.05	Process PID out	Output of the process PID controller.	10 = 1 unit
04.06	Process var1	Process variable 1. See parameter group 35 Process variable .	1000 = 1
04.07	Process var2	Process variable 2. See parameter group 35 Process variable .	1000 = 1
04.08	Process var3	Process variable 3. See parameter group 35 Process variable .	1000 = 1
04.09	Counter ontime1	Reading of on-time counter 1. See parameter 44.01 Ontime1 func . Can be reset by entering a 0.	1 = 1 s
04.10	Counter ontime2	Reading of on-time counter 2. See parameter group 44.05 Ontime2 func . Can be reset by entering a 0.	1 = 1 s
04.11	Counter edge1	Reading of rising edge counter 1. See parameter group 44.09 Edge count1 func . Can be reset by entering a 0.	1 = 1
04.12	Counter edge2	Reading of rising edge counter 2. See parameter group 44.14 Edge count2 func . Can be reset by entering a 0.	1 = 1
04.13	Counter value1	Reading of value counter 1. See parameter group 44.19 Val count1 func . Can be reset by entering a 0.	1 = 1
04.14	Counter value2	Reading of value counter 2. See parameter group 44.24 Val count2 func . Can be reset by entering a 0.	1 = 1

No.	Name/Value	Description	FbEq
06 Drive status		Drive status words.	
06.01	Status word1	Status word 1 of the drive.	-
Bit	Name	Information	
0	Ready	1 = Drive is ready to receive start command. 0 = Drive is not ready.	
1	Enabled	1 = External run enable signal is received. 0 = No external run enable signal is received.	
2	Started	1 = Drive has received start command. 0 = Drive has not received start command.	
3	Running	1 = Drive is modulating. 0 = Drive is not modulating.	
4	Em off (off2)	1 = Emergency OFF2 is active. 0 = Emergency OFF2 is inactive.	
5	Em stop (off3)	1 = Emergency OFF3 (ramp stop) is active. 0 = Emergency OFF3 is inactive.	
6	Ack startinh	1 = Start inhibit is active. 0 = Start inhibit is inactive.	
7	Alarm	1 = Alarm is active. See chapter Fault tracing on page 283. 0 = No alarm is active.	
8	Ext2 act	1 = External control EXT2 is active. 0 = External control EXT1 is active.	
9	Local fb	1 = Fieldbus local control is active. 0 = Fieldbus local control is inactive.	
10	Fault	1 = Fault is active. See chapter Fault tracing on page 283. 0 = No fault is active.	
11	Local panel	1 = Local control is active, ie. drive is controlled from PC tool or control panel. 0 = Local control is inactive.	
12	Fault(-1)	1 = No fault is active. 0 = Fault is active. See chapter Fault tracing on page 283.	
13...31	Reserved		

No.	Name/Value	Description	FbEq
06.02	Status word2	Status word 2 of the drive.	-
	Bit	Name	Information
	0	Start act	1 = Drive start command is active. 0 = Drive start command is inactive.
	1	Stop act	1 = Drive stop command is active. 0 = Drive stop command is inactive.
	2	Ready relay	1 = Ready to function: run enable signal on, no fault, emergency stop signal off, no ID run inhibition. Connected by default to DIO1 by par. 14.03 DIO1 out src . 0 = Not ready to function.
	3	Modulating	1 = Modulating: IGBTs are controlled, ie. the drive is RUNNING. 0 = No modulation: IGBTs are not controlled.
	4	Ref running	1 = Normal operation is enabled. Running. Drive follows the given reference. 0 = Normal operation is disabled. Drive is not following the given reference (eg. in magnetization phase drive is modulating).
	5	Jogging	1 = Jogging function 1 or 2 is active. 0 = Jogging function is inactive.
	6	Off1	1 = Emergency stop OFF1 is active. 0 = Emergency stop OFF1 is inactive.
	7	Start inh mask	1 = Maskable (by par. 12.01 Start inhibit) start inhibit is active. 0 = No maskable start inhibit is active.
	8	Start inh nomask	1 = Non-maskable start inhibit is active. 0 = No non-maskable start inhibit is active.
	9	Chrg rel closed	1 = Charging relay is closed. 0 = Charging relay is open.
	10	Sto act	1 = Safe torque off function is active. See parameter 30.07 Sto diagnostic . 0 = Safe torque off function is inactive.
	11	Reserved	
	12	Ramp in 0	1 = Ramp Function Generator input is forced to zero. 0 = Normal operation.
	13	Ramp hold	1 = Ramp Function Generator output is held. 0 = Normal operation.
	14	Ramp out 0	1 = Ramp Function Generator output is forced to zero. 0 = Normal operation.
	15...31	Reserved	

No.	Name/Value	Description	FbEq
06.03	Speed ctrl stat	Speed control status word.	-
	Bit	Name	Information
	0	Speed act neg	1 = Actual speed is negative.
	1	Zero speed	1 = Actual speed has reached the zero speed limit (parameters 19.06 Zero speed limit and 19.07 Zero speed delay).
	2	Above limit	1 = Actual speed has exceeded the supervision limit (parameter 19.08 Above speed lim).
	3	At setpoint	1 = The difference between the actual speed and the unramped speed reference is within the speed window (parameter 19.10 Speed window).
	4	Reserved	
	5	PI tune active	1 = Speed controller autotune is active.
	6	PI tune request	1 = Speed controller autotune has been requested by parameter 23.20 PI tune mode .
	7	PI tune done	1 = Speed controller autotune has been completed successfully.
	8	Speed not zero	1 = Speed controller autotune has been requested when the drive was running, but zero speed has not been reached within preset maximum time.
	9	Spd tune aborted	1 = Speed controller autotune has been aborted by a stop command.
	10	Spd tune timeout	1 = A speed controller autotune timeout has occurred. <ul style="list-style-type: none"> Autotune was requested while drive was running, but a stop command did not follow Stop command has been issued, but drive has not reached zero speed Drive does not accelerate and decelerate according to the given reference during autotune.
06.05	Limit word1	Limit word 1.	-
	Bit	Name	Information
	0	Torq lim	1 = Drive torque is being limited by the motor control (undervoltage control, current control, load angle control or pull-out control), or by the torque limit parameters in group 20 Limits .
	1	Spd ctl tlim min	1 = Speed controller output minimum torque limit is active. The limit is defined by parameter 23.10 Min torq sp ctrl .
	2	Spd ctl tlim max	1 = Speed controller output maximum torque limit is active. The limit is defined by parameter 23.09 Max torq sp ctrl .
	3	Torq ref max	1 = Torque reference (03.11 Torq ref ramped) maximum limit is active. The limit is defined by parameter 24.03 Maximum torq ref .
	4	Torq ref min	1 = Torque reference (03.11 Torq ref ramped) minimum limit is active. The limit is defined by parameter 24.04 Minimum torq ref .
	5	Tlim max speed	1 = Torque reference maximum value is limited by the rush control, because of maximum speed limit 20.01 Maximum speed .
	6	Tlim min speed	1 = Torque reference minimum value is limited by the rush control, because of minimum speed limit 20.02 Minimum speed .

No.	Name/Value	Description	FbEq
06.07	Torq lim status	Torque controller limitation status word.	-
	Bit	Name	Information
	0	Undervoltage	1 = Intermediate circuit DC undervoltage. *
	1	Overvoltage	1 = Intermediate circuit DC overvoltage. *
	2	Minimum torque	1 = Torque reference minimum limit is active. The limit is defined by parameter 24.04 Minimum torq ref. *
	3	Maximum torque	1 = Torque reference maximum limit is active. The limit is defined by parameter 24.03 Maximum torq ref. *
	4	Internal current	1 = An inverter current limit is active. The limit is identified by bits 8...11.
	5	Load angle	1 = For permanent magnet motor only: Load angle limit is active, i.e. the motor cannot produce more torque.
	6	Motor pull-out	1 = For asynchronous motor only: Motor pull-out limit is active, i.e. the motor cannot produce more torque.
	7	Reserved	
	8	Thermal	1 = Input current is limited by main circuit thermal limit.
	9	INU maximum	1 = Inverter maximum output current limit is active (limits the drive output current I_{MAX}). **
	10	User current	1 = Maximum inverter output current limit is active. The limit is defined by parameter 20.05 Maximum current. **
	11	Thermal IGBT	1 = Calculated thermal current value limits the inverter output current. **
	* One of bits 0...3 can be on simultaneously. The bit typically indicates the limit that is exceeded first.		
	** Only one of bits 9...11 can be on simultaneously. The bit typically indicates the limit that is exceeded first.		
06.12	Op mode ack	Operation mode acknowledge: 0 = Stopped, 1 = Speed, 2 = Torque, 3 = Min, 4 = Max, 5 = Add, 10 = Scalar, 11 = Forced Magn (i.e. DC Hold)	1 = 1
06.13	Superv status	Supervision status word. Bits 0...2 reflect the status of supervisory functions 1...3 respectively. The functions are configured in parameter group 33 Supervision (page 198).	-
06.14	Timed func stat	Bits 0...3 show the on/off status of the four timers (1...4 respectively) configured in parameter group 36 Timed functions (page 209). Bit 4 is on if any one of the four timers is on.	-
06.15	Counter status	Counter status word. Shows whether the maintenance counters configured in parameter group 44 Maintenance (page 221) have exceeded their limits.	-
	Bit	Name	Information
	0	Ontime1	1 = On-time counter 1 has reached its preset limit.
	1	Ontime2	1 = On-time counter 2 has reached its preset limit.
	2	Edge1	1 = Rising edge counter 1 has reached its preset limit.
	3	Edge2	1 = Rising edge counter 2 has reached its preset limit.
	4	Value1	1 = Value counter 1 has reached its preset limit.
	5	Value2	1 = Value counter 2 has reached its preset limit.
08 Alarms & faults		Alarm and fault information.	
08.01	Active fault	Fault code of the latest fault.	1 = 1
08.02	Last fault	Fault code of the 2nd latest fault.	1 = 1

No.	Name/Value	Description	FbEq																																		
08.03	Fault time hi	Time (real time or power-on time) at which the active fault occurred in format dd.mm.yy (day, month and year).	1 = 1 d																																		
08.04	Fault time lo	Time (real time or power-on time) at which the active fault occurred in format hh.mm.ss (hours, minutes and seconds).	1 = 1																																		
08.05	Alarm word1	Alarm word 1. For possible causes and remedies, see chapter Fault tracing on page 283. Can be reset by entering a 0.	-																																		
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Name</th> </tr> </thead> <tbody> <tr><td>0</td><td>Brake start torq</td></tr> <tr><td>1</td><td>Brake not closed</td></tr> <tr><td>2</td><td>Brake not open</td></tr> <tr><td>3</td><td>Safe torq off</td></tr> <tr><td>4</td><td>Sto mode</td></tr> <tr><td>5</td><td>Motor temp1</td></tr> <tr><td>6</td><td>Em off</td></tr> <tr><td>7</td><td>Run enable</td></tr> <tr><td>8</td><td>Id run</td></tr> <tr><td>9</td><td>Em stop</td></tr> <tr><td>10</td><td>Position scaling</td></tr> <tr><td>11</td><td>Br overtemp</td></tr> <tr><td>12</td><td>BC igbt overtemp</td></tr> <tr><td>13</td><td>Device overtemp</td></tr> <tr><td>14</td><td>Int board ovtemp</td></tr> <tr><td>15</td><td>BC mod overtemp</td></tr> </tbody> </table>	Bit	Name	0	Brake start torq	1	Brake not closed	2	Brake not open	3	Safe torq off	4	Sto mode	5	Motor temp1	6	Em off	7	Run enable	8	Id run	9	Em stop	10	Position scaling	11	Br overtemp	12	BC igbt overtemp	13	Device overtemp	14	Int board ovtemp	15	BC mod overtemp	
Bit	Name																																				
0	Brake start torq																																				
1	Brake not closed																																				
2	Brake not open																																				
3	Safe torq off																																				
4	Sto mode																																				
5	Motor temp1																																				
6	Em off																																				
7	Run enable																																				
8	Id run																																				
9	Em stop																																				
10	Position scaling																																				
11	Br overtemp																																				
12	BC igbt overtemp																																				
13	Device overtemp																																				
14	Int board ovtemp																																				
15	BC mod overtemp																																				
08.06	Alarm word2	Alarm word 2. For possible causes and remedies, see chapter Fault tracing on page 283. Can be reset by entering a 0.	-																																		
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Name</th> </tr> </thead> <tbody> <tr><td>0</td><td>Inu overtemp</td></tr> <tr><td>1</td><td>FBA comm</td></tr> <tr><td>2</td><td>Panel loss</td></tr> <tr><td>3</td><td>AI supervision</td></tr> <tr><td>4</td><td>FBA par conf</td></tr> <tr><td>5</td><td>No motor data</td></tr> <tr><td>6</td><td>Encoder1</td></tr> <tr><td>7</td><td>Encoder2</td></tr> <tr><td>8</td><td>Latch pos1</td></tr> <tr><td>9</td><td>Latch pos2</td></tr> <tr><td>10</td><td>Enc emul</td></tr> <tr><td>11</td><td>FEN temp meas</td></tr> <tr><td>12</td><td>Emul max freq</td></tr> <tr><td>13</td><td>Emul pos ref</td></tr> <tr><td>14</td><td>Resolver atune</td></tr> <tr><td>15</td><td>Enc1 cable</td></tr> </tbody> </table>	Bit	Name	0	Inu overtemp	1	FBA comm	2	Panel loss	3	AI supervision	4	FBA par conf	5	No motor data	6	Encoder1	7	Encoder2	8	Latch pos1	9	Latch pos2	10	Enc emul	11	FEN temp meas	12	Emul max freq	13	Emul pos ref	14	Resolver atune	15	Enc1 cable	
Bit	Name																																				
0	Inu overtemp																																				
1	FBA comm																																				
2	Panel loss																																				
3	AI supervision																																				
4	FBA par conf																																				
5	No motor data																																				
6	Encoder1																																				
7	Encoder2																																				
8	Latch pos1																																				
9	Latch pos2																																				
10	Enc emul																																				
11	FEN temp meas																																				
12	Emul max freq																																				
13	Emul pos ref																																				
14	Resolver atune																																				
15	Enc1 cable																																				

No.	Name/Value	Description	FbEq
08.07	Alarm word3	Alarm word 3. For possible causes and remedies, see chapter Fault tracing on page 283. Can be reset by entering a 0.	-
	Bit	Name	
	0	Enc2 cable	
	1	D2D comm	
	2	D2D buffer ol	
	3	PS comm	
	4	Restore	
	5	Curr meas calib	
	6	Autophasing	
	7	Earthfault	
	8	Autoreset	
	9	Motor nom value	
	10	D2D config	
	11	Stall	
	12	Load curve	
	13	Load curve conf	
	14	U/f curve conf	
	15	Speed meas	
08.08	Alarm word4	Alarm word 4. For possible causes and remedies, see chapter Fault tracing on page 283. Can be reset by entering a 0.	-
	Bit	Name	
	0	Option comm loss	
	1	Solution prog	
	2	Motor temp2	
	3	IGBT overload	
	4	IGBT temp	
	5	Cooling	
	6	Menu changed	
	7	Temp meas fail	
	8	Mnt counter (common for maintenance counter alarms 2066...2071)	
	9	DC not charged	
	10	Speed tune fail	
	11	Start interlock	
09 System info		Drive type, program revision and option slot occupation information.	
09.01	Drive type	Displays the drive type (for example, ACS850).	-

No.	Name/Value	Description	FbEq						
09.02	Drive rating ID	Displays the inverter type (ACS850-xx-...) of the drive. 0 = Unconfigured, 101 = 03A0, 102 = 03A6, 103 = 04A8, 104 = 06A0, 105 = 08A0, 106 = 010A, 107 = 014A, 108 = 018A, 109 = 025A, 110 = 030A, 111 = 035A, 112 = 044A, 113 = 050A, 114 = 061A, 115 = 078A, 116 = 094A, 117 = 103A, 118 = 144A, 119 = 166A, 120 = 202A, 121 = 225A, 122 = 260A, 123 = 290A, 124 = 430A, 125 = 521A, 126 = 602A, 127 = 693A, 128 = 720A, 141 = 03A0_2, 142 = 03A6_2, 143 = 04A8_2, 144 = 06A0_2, 145 = 08A0_2, 146 = 010A_2, 147 = 014A_2, 148 = 018A_2, 149 = 025A_2, 150 = 030A_2, 151 = 035A_2, 152 = 044A_2, 153 = 050A_2, 154 = 061A_2, 155 = 078A_2, 156 = 094A_2	1 = 1						
09.03	Firmware ID	Displays the firmware name. E.g. UIFL.	-						
09.04	Firmware ver	Displays the version of the firmware package in the drive, e.g. E00F hex.	-						
09.05	Firmware patch	Displays the version of the firmware patch in the drive.	1 = 1						
09.10	Int logic ver	Displays the version of the logic on the main circuit board of the drive.	-						
09.20	Option slot1	Displays the type of the optional module in option slot 1. 0 = No option, 1 = No comm, 2 = Unknown, 3 = FEN-01, 4 = FEN-11, 5 = FEN-21, 6 = FIO-01, 7 = FIO-11, 8 = FPBA-01, 9 = FPBA-02, 10 = FCAN-01, 11 = FDNA-01, 12 = FENA-01, 13 = FENA-02, 14 = FLON-01, 15 = FRSA-00, 16 = FMBA-01, 17 = FFOA-01, 18 = FFOA-02, 19 = FSEN-21, 20 = FEN-31, 21 = FIO-21, 22 = FSCA-01, 23 = FSEA-21, 24 = FIO-31, 25 = FECA-01	1 = 1						
09.21	Option slot2	Displays the type of the optional module in option slot 2. See signal 09.20 Option slot1 .	1 = 1						
09.22	Option slot3	Displays the type of the optional module in option slot 3. See signal 09.20 Option slot1 .	1 = 1						
10 Start/stop/dir		Start/stop/direction etc. signal source selections.							
10.01	Ext1 start func	Selects the source of start and stop commands for external control location 1 (EXT1). Note: This parameter cannot be changed while the drive is running.							
	Not sel	No start or stop command sources selected.	0						
	In1	The source of the start and stop commands is selected by parameter 10.02 Ext1 start in1 . The state transitions of the source bit are interpreted as follows: <table border="1" data-bbox="473 1671 921 1805"> <thead> <tr> <th>State of source (via par 10.02)</th> <th>Command</th> </tr> </thead> <tbody> <tr> <td>0 -> 1</td> <td>Start</td> </tr> <tr> <td>1 -> 0</td> <td>Stop</td> </tr> </tbody> </table>	State of source (via par 10.02)	Command	0 -> 1	Start	1 -> 0	Stop	1
State of source (via par 10.02)	Command								
0 -> 1	Start								
1 -> 0	Stop								

No.	Name/Value	Description	FbEq															
	3-wire	<p>The sources of the start and stop commands is selected by parameters 10.02 Ext1 start in1 and 10.03 Ext1 start in2. The state transitions of the source bits are interpreted as follows:</p> <table border="1"> <thead> <tr> <th>State of source 1 (via par. 10.02)</th> <th>State of source 2 (via par. 10.03)</th> <th>Command</th> </tr> </thead> <tbody> <tr> <td>0 -> 1</td> <td>1</td> <td>Start</td> </tr> <tr> <td>Any</td> <td>1 -> 0</td> <td>Stop</td> </tr> <tr> <td>Any</td> <td>0</td> <td>Stop</td> </tr> </tbody> </table>	State of source 1 (via par. 10.02)	State of source 2 (via par. 10.03)	Command	0 -> 1	1	Start	Any	1 -> 0	Stop	Any	0	Stop	2			
State of source 1 (via par. 10.02)	State of source 2 (via par. 10.03)	Command																
0 -> 1	1	Start																
Any	1 -> 0	Stop																
Any	0	Stop																
	FB	The start and stop commands are taken from the fieldbus Control Word which is defined by parameter 50.15 Fb cw used .	3															
	D2D	The start and stop commands are taken from another drive through the D2D (Drive-to-drive) Control Word.	4															
	In1F In2R	<p>The source selected by 10.02 Ext1 start in1 is the forward start signal, the source selected by 10.03 Ext1 start in2 is the reverse start signal.</p> <table border="1"> <thead> <tr> <th>State of source 1 (via par. 10.02)</th> <th>State of source 2 (via par. 10.03)</th> <th>Command</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Stop</td> </tr> <tr> <td>1</td> <td>0</td> <td>Start forward</td> </tr> <tr> <td>0</td> <td>1</td> <td>Start reverse</td> </tr> <tr> <td>1</td> <td>1</td> <td>Stop</td> </tr> </tbody> </table>	State of source 1 (via par. 10.02)	State of source 2 (via par. 10.03)	Command	0	0	Stop	1	0	Start forward	0	1	Start reverse	1	1	Stop	5
State of source 1 (via par. 10.02)	State of source 2 (via par. 10.03)	Command																
0	0	Stop																
1	0	Start forward																
0	1	Start reverse																
1	1	Stop																
	In1St In2Dir	The source selected by 10.02 Ext1 start in1 is the start signal (0 = stop, 1 = start), the source selected by 10.03 Ext1 start in2 is the direction signal (0 = forward, 1 = reverse).	6															
10.02	Ext1 start in1	<p>Selects source 1 of start and stop commands for external control location EXT1. See parameter 10.01 Ext1 start func, selections In1 and 3-wire.</p> <p>Note: This parameter cannot be changed while the drive is running.</p>																
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337															
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017															
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947															
	Timed func	Bit 4 of parameter 06.14 Timed func stat . The bit is on when at least one of the four timers configured in parameter group 36 Timed functions is on.	1074005518															
	Const	Constant and bit pointer settings (see Terms and abbreviations on page 102).	-															
	Pointer																	
10.03	Ext1 start in2	<p>Selects source 2 of start and stop commands for external control location EXT1. See parameter 10.01 Ext1 start func, selection 3-wire.</p> <p>Note: This parameter cannot be changed while the drive is running.</p>																
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873															
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481															
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483															

No.	Name/Value	Description	FbEq															
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-															
	Pointer																	
10.04	Ext2 start func	Selects the source of start and stop commands for external control location 2 (EXT2). Note: This parameter cannot be changed while the drive is running.																
	Not sel	No start or stop command sources selected.	0															
	In1	The source of the start and stop commands is selected by parameter 10.05 Ext2 start in1 . The state transitions of the source bit are interpreted as follows: <table border="1" data-bbox="475 633 921 768"> <thead> <tr> <th>State of source (via par. 10.05)</th> <th>Command</th> </tr> </thead> <tbody> <tr> <td>0 -> 1</td> <td>Start</td> </tr> <tr> <td>1 -> 0</td> <td>Stop</td> </tr> </tbody> </table>	State of source (via par. 10.05)	Command	0 -> 1	Start	1 -> 0	Stop	1									
State of source (via par. 10.05)	Command																	
0 -> 1	Start																	
1 -> 0	Stop																	
	3-wire	The sources of the start and stop commands is selected by parameters 10.05 Ext2 start in1 and 10.06 Ext2 start in2 . The state transitions of the source bits are interpreted as follows: <table border="1" data-bbox="475 925 1188 1093"> <thead> <tr> <th>State of source 1 (via par. 10.05)</th> <th>State of source 2 (via par. 10.06)</th> <th>Command</th> </tr> </thead> <tbody> <tr> <td>0 -> 1</td> <td>1</td> <td>Start</td> </tr> <tr> <td>Any</td> <td>1 -> 0</td> <td>Stop</td> </tr> <tr> <td>Any</td> <td>0</td> <td>Stop</td> </tr> </tbody> </table>	State of source 1 (via par. 10.05)	State of source 2 (via par. 10.06)	Command	0 -> 1	1	Start	Any	1 -> 0	Stop	Any	0	Stop	2			
State of source 1 (via par. 10.05)	State of source 2 (via par. 10.06)	Command																
0 -> 1	1	Start																
Any	1 -> 0	Stop																
Any	0	Stop																
	FB	The start and stop commands are taken from the fieldbus Control Word which is defined by parameter 50.15 Fb cw used .	3															
	D2D	The start and stop commands are taken from another drive through the D2D (Drive-to-drive) Control Word.	4															
	In1F In2R	The source selected by 10.05 Ext2 start in1 is the forward start signal, the source selected by 10.06 Ext2 start in2 is the reverse start signal. <table border="1" data-bbox="475 1440 1188 1641"> <thead> <tr> <th>State of source 1 (via par. 10.05)</th> <th>State of source 2 (via par. 10.06)</th> <th>Command</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Stop</td> </tr> <tr> <td>1</td> <td>0</td> <td>Start forward</td> </tr> <tr> <td>0</td> <td>1</td> <td>Start reverse</td> </tr> <tr> <td>1</td> <td>1</td> <td>Stop</td> </tr> </tbody> </table>	State of source 1 (via par. 10.05)	State of source 2 (via par. 10.06)	Command	0	0	Stop	1	0	Start forward	0	1	Start reverse	1	1	Stop	5
State of source 1 (via par. 10.05)	State of source 2 (via par. 10.06)	Command																
0	0	Stop																
1	0	Start forward																
0	1	Start reverse																
1	1	Stop																
	In1St In2Dir	The source selected by 10.05 Ext2 start in1 is the start signal (0 = stop, 1 = start), the source selected by 10.06 Ext2 start in2 is the direction signal (0 = forward, 1 = reverse).	6															
10.05	Ext2 start in1	Selects source 1 of start and stop commands for external control location EXT2. See parameter 10.04 Ext2 start func , selections In1 and 3-wire . Note: This parameter cannot be changed while the drive is running.																
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337															
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017															
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947															



No.	Name/Value	Description	FbEq
	Timed func	Bit 4 of parameter 06.14 Timed func stat . The bit is on when any one of the four timers configured in parameter group 36 Timed functions is on.	1074005518
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
10.06	Ext2 start in2	Selects source 2 of start and stop commands for external control location EXT2. See parameter 10.04 Ext2 start func , selection 3-wire . Note: This parameter cannot be changed while the drive is running.	
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
10.07	Jog1 start	If enabled by parameter 10.09 Jog enable , selects the source for the activation of jogging function 1. (Jogging function 1 can also be activated through fieldbus regardless of parameter 10.09 .) 1 = Active. See also other jogging function parameters: 10.08 Jog2 start , 10.09 Jog enable , 21.07 Speed ref jog1 , 21.08 Speed ref jog2 , 22.10 Acc time jogging , 22.11 Dec time jogging and 19.07 Zero speed delay . Note: This parameter cannot be changed while the drive is running.	
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
10.08	Jog2 start	If enabled by parameter 10.09 Jog enable , selects the source for the activation of jogging function 2. (Jogging function 2 can also be activated through fieldbus regardless of parameter 10.09 .) 1 = Active. See also parameter 10.07 Jog1 start . Note: This parameter cannot be changed while the drive is running.	
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017

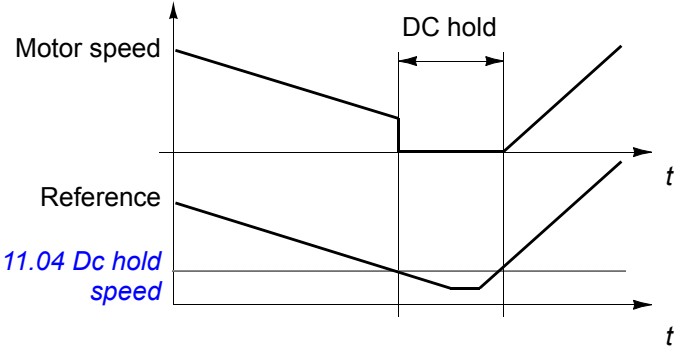
No.	Name/Value	Description	FbEq
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
10.09	Jog enable	Selects the source for enabling parameters 10.07 Jog1 start and 10.08 Jog2 start . Note: Jogging can be enabled using this parameter only when no start command from an external control location is active. On the other hand, if jogging is already enabled, the drive cannot be started from an external control location apart from jog commands through fieldbus.	
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
10.10	Fault reset sel	Selects the source of the external fault reset signal. The signal resets the drive after a fault trip if the cause of the fault no longer exists. 0 -> 1 = Fault reset. Note: A fault reset from the fieldbus is always observed regardless of this setting.	
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		

No.	Name/Value	Description	FbEq
10.11	Run enable	Selects the source of the external run enable signal. If the run enable signal is switched off, the drive will not start, or coasts to stop if running. 1 = Run enable. Note: This parameter cannot be changed while the drive is running.	
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019
	COMM.CW	External signal required through the fieldbus Control Word (as indicated by 02.22 FBA main cw , bit 7).	1074201122
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
10.13	Em stop off3	Selects the source of the emergency stop OFF3 signal. The drive is stopped along the emergency stop ramp time defined by parameter 22.12 Em stop time . 0 = OFF3 active. Note: This parameter cannot be changed while the drive is running.	
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		

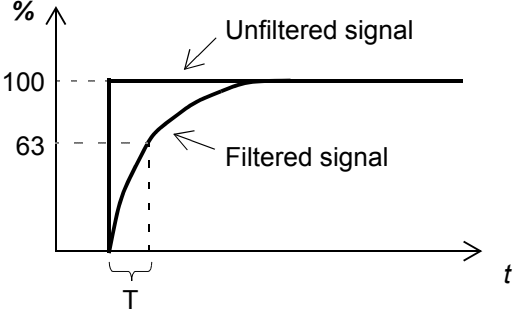
No.	Name/Value	Description	FbEq
10.15	Em stop off1	Selects the source of the emergency stop OFF1 signal. The drive is stopped using the active deceleration time. Emergency stop can also be activated through fieldbus (02.22 FBA main cw or 02.36 EFB main cw). 0 = OFF1 active. Note: This parameter cannot be changed while the drive is running.	
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
10.17	Start enable	Selects the source for the start enable signal. 1 = Start enable. If the signal is switched off, the drive will not start or coasts to stop if running.	
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		

No.	Name/Value	Description	FbEq
10.19	Start inhibit	Enables the start inhibit function. The function prevents drive restart (i.e. protects against unexpected start) if <ul style="list-style-type: none"> • the drive trips on a fault and the fault is reset, • the run enable signal is activated while the start command is active (see parameter 10.11 Run enable), • control changes from local to remote, or • external control switches from EXT1 to EXT2 or vice versa. A new rising edge of the start command is needed after the start inhibit has been activated. Note that in certain applications it is necessary to allow the drive to restart.	
	Disabled	The start inhibit function is disabled.	0
	Enabled	The start inhibit function is enabled.	1
10.20	Start interl func	Defines how the start interlock input (DIIL) on the JCU control unit affects the drive operation.	
	Off2 stop	With the drive running: <ul style="list-style-type: none"> • 1 = Normal operation. • 0 = Stop by coasting. The drive can be restarted by restoring the start interlock signal and switching the start signal from 0 to 1. With the drive stopped: <ul style="list-style-type: none"> • 1 = Starting allowed. • 0 = Starting not allowed. 	0
	Off3 stop	With the drive running: <ul style="list-style-type: none"> • 1 = Normal operation. • 0 = Stop by ramping. The deceleration time is defined by parameter 22.12 Em stop time. The drive can be restarted by restoring the start interlock signal and switching the start signal from 0 to 1. With the drive stopped: <ul style="list-style-type: none"> • 1 = Starting allowed. • 0 = Starting not allowed. 	1
11 Start/stop mode		Start, stop, magnetization etc. settings.	
11.01	Start mode	Selects the motor start function. Notes: <ul style="list-style-type: none"> • Selections <i>Fast</i> and <i>Const time</i> are ignored if parameter 99.05 is set to <i>Scalar</i>. • Starting to a rotating machine is not possible when DC magnetizing is selected (<i>Fast</i> or <i>Const time</i>). • With permanent magnet motors, <i>Automatic</i> start must be used. 	
	Fast	The drive pre-magnetizes the motor before start. The pre-magnetizing time is determined automatically, being typically 200 ms to 2 s depending on motor size. This mode should be selected if a high break-away torque is required. Note: This parameter cannot be changed while the drive is running.	0

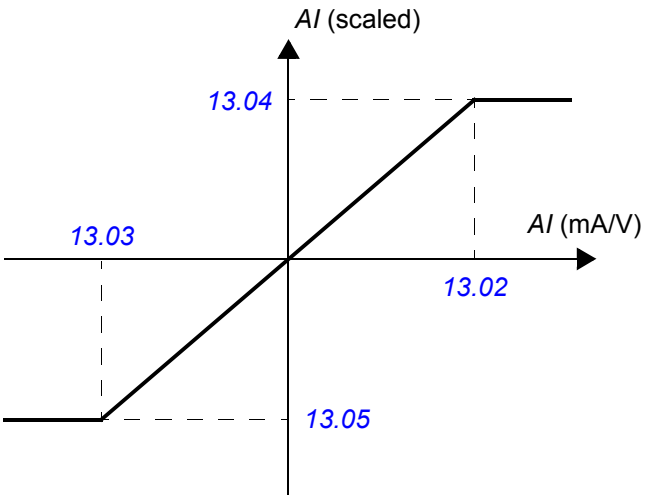
No.	Name/Value	Description	FbEq										
	Const time	<p>The drive pre-magnetizes the motor before start. The pre-magnetizing time is defined by parameter 11.02 Dc-magn time. This mode should be selected if constant pre-magnetizing time is required (e.g. if the motor start must be synchronized with the release of a mechanical brake). This setting also guarantees the highest possible break-away torque when the pre-magnetizing time is set long enough.</p> <p> WARNING! The drive will start after the set magnetizing time has passed even if motor magnetization is not completed. In applications where a full break-away torque is essential, ensure that the constant magnetizing time is long enough to allow generation of full magnetization and torque.</p>	1										
	Automatic	<p>Automatic start guarantees optimal motor start in most cases. It includes the flying start function (starting to a rotating machine) and the automatic restart function (a stopped motor can be restarted immediately without waiting the motor flux to die away). The drive motor control program identifies the flux as well as the mechanical state of the motor and starts the motor instantly under all conditions.</p> <p>Note: If parameter 99.05 Motor ctrl mode is set to <i>Scalar</i>, no flying start or automatic restart is possible by default.</p>	2										
11.02	Dc-magn time	<p>Defines the constant DC magnetizing time. See parameter 11.01 Start mode. After the start command, the drive automatically premagnetizes the motor the set time. To ensure full magnetizing, set this value to the same value as or higher than the rotor time constant. If not known, use the rule-of-thumb value given in the table below:</p> <table border="1" data-bbox="475 1151 1185 1397"> <thead> <tr> <th>Motor rated power</th> <th>Constant magnetizing time</th> </tr> </thead> <tbody> <tr> <td>< 1 kW</td> <td>≥ 50 to 100 ms</td> </tr> <tr> <td>1 to 10 kW</td> <td>≥ 100 to 200 ms</td> </tr> <tr> <td>10 to 200 kW</td> <td>≥ 200 to 1000 ms</td> </tr> <tr> <td>200 to 1000 kW</td> <td>≥ 1000 to 2000 ms</td> </tr> </tbody> </table> <p>Note: This parameter cannot be changed while the drive is running.</p>	Motor rated power	Constant magnetizing time	< 1 kW	≥ 50 to 100 ms	1 to 10 kW	≥ 100 to 200 ms	10 to 200 kW	≥ 200 to 1000 ms	200 to 1000 kW	≥ 1000 to 2000 ms	
Motor rated power	Constant magnetizing time												
< 1 kW	≥ 50 to 100 ms												
1 to 10 kW	≥ 100 to 200 ms												
10 to 200 kW	≥ 200 to 1000 ms												
200 to 1000 kW	≥ 1000 to 2000 ms												
	0 ... 10000 ms	Constant DC magnetizing time.	1 = 1 ms										
11.03	Stop mode	Selects the motor stop function.											
	Coast	<p>Stop by cutting of the motor power supply. The motor coasts to a stop.</p> <p> WARNING! If the mechanical brake is used, ensure it is safe to stop the drive by coasting.</p>	1										
	Ramp	Stop along ramp. See parameter group 22 Speed ref ramp on page 170 .	2										
11.04	Dc hold speed	Defines the DC hold speed. See parameter 11.06 Dc hold .											
	0.0 ... 1000.0 rpm	DC hold speed.	10 = 1 rpm										
11.05	Dc hold curr ref	Defines the DC hold current in percent of the motor nominal current. See parameter 11.06 Dc hold .											
	0 ... 100%	DC hold current.	1 = 1%										

No.	Name/Value	Description	FbEq
11.06	Dc hold	<p>Enables the DC hold function. The function makes it possible to lock the rotor at zero speed.</p> <p>When both the reference and the speed drop below the value of parameter 11.04 Dc hold speed, the drive will stop generating sinusoidal current and start to inject DC into the motor. The current is set by parameter 11.05 Dc hold curr ref. When the reference speed exceeds parameter 11.04 Dc hold speed, normal drive operation continues.</p>  <p>Notes:</p> <ul style="list-style-type: none"> • The DC hold function has no effect if the start signal is switched off. • The DC hold function can only be activated in speed control mode. • The DC hold function cannot be activated if parameter 99.05 Motor ctrl mode is set to <i>Scalar</i>. • Injecting DC current into the motor causes the motor to heat up. In applications where long DC hold times are required, externally ventilated motors should be used. If the DC hold period is long, the DC hold cannot prevent the motor shaft from rotating if a constant load is applied to the motor. 	
	Disabled	The DC hold function is disabled.	0
	Enabled	The DC hold function is enabled.	1
11.07	Autophasing mode	Selects the way autophasing is performed during the ID run. See section Autophasing on page 67.	
	Turning	This mode gives the most accurate autophasing result. This mode can be used, and is recommended, if it is allowed for the motor to rotate during the ID run and the start-up is not time-critical. Note: This mode will cause the motor to rotate during the ID run.	0
	Standstill 1	Faster than the Turning mode, but not as accurate. The motor will not rotate.	1
	Standstill 2	An alternative standstill autophasing mode that can be used if the Turning mode cannot be used, and the Standstill 1 mode gives erratic results. However, this mode is considerably slower than Standstill 1 .	2
12 Operating mode		Selection of external control location and operating modes.	
12.01	Ext1/Ext2 sel	Selects the source for external control location EXT1/EXT2 selection. 0 = EXT1 1 = EXT2	

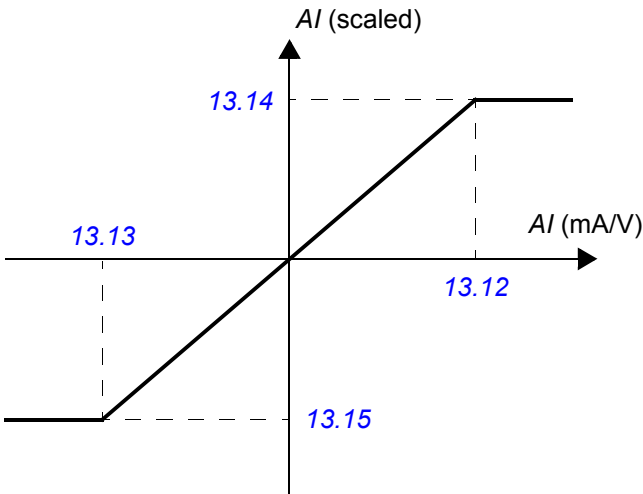
No.	Name/Value	Description	FbEq
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
12.03	Ext1 ctrl mode	Selects the operating mode for external control location EXT1.	
	Speed	Speed control. The speed controller output (torque reference) is 03.09 Torq ref sp ctrl .	1
	Torque	Torque control. Torque reference is 03.12 Torq ref sp lim .	2
	Min	Combination of selections Speed and Torque : Torque selector compares the torque reference and the speed controller output and the smaller of the two is used.	3
	Max	Combination of selections Speed and Torque : Torque selector compares the torque reference and the speed controller output and the greater of the two is used.	4
	Add	Combination of selections Speed and Torque : Torque selector adds the speed controller output to the torque reference.	5
12.05	Ext2 ctrl mode	Selects the operating mode for external control location EXT2.	
	Speed	Speed control. The speed controller output (torque reference) is 03.09 Torq ref sp ctrl .	1
	Torque	Torque control. Torque reference is 03.12 Torq ref sp lim .	2
	Min	Combination of selections Speed and Torque : Torque selector compares the torque reference and the speed controller output and the smaller of the two is used.	3
	Max	Combination of selections Speed and Torque : Torque selector compares the torque reference and the speed controller output and the greater of the two is used.	4
	Add	Combination of selections Speed and Torque : Torque selector adds the speed controller output to the torque reference.	5
12.07	Local ctrl mode	Selects the operating mode for local control.	
	Speed	Speed control. Torque reference is 03.09 Torq ref sp ctrl .	1
	Torque	Torque control. Torque reference is 03.12 Torq ref sp lim .	2

No.	Name/Value	Description	FbEq
13 Analogue inputs		Analog input signal processing.	
13.01	AI1 filt time	<p>Defines the filter time constant for analogue input AI1.</p>  $O = I \times (1 - e^{-t/T})$ <p> <i>I</i> = filter input (step) <i>O</i> = filter output <i>t</i> = time <i>T</i> = filter time constant </p> <p>Note: The signal is also filtered due to the signal interface hardware (approximately 0.25 ms time constant). This cannot be changed by any parameter.</p>	
	0.000 ... 30.000 s	Filter time constant.	1000 = 1 s
13.02	AI1 max	Defines the maximum value for analogue input AI1. The input type is selected with jumper J1 on the JCU Control Unit.	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	Maximum AI1 value.	1000 = 1 unit
13.03	AI1 min	Defines the minimum value for analogue input AI1. The input type is selected with jumper J1 on the JCU Control Unit.	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	Minimum AI1 value.	1000 = 1 unit

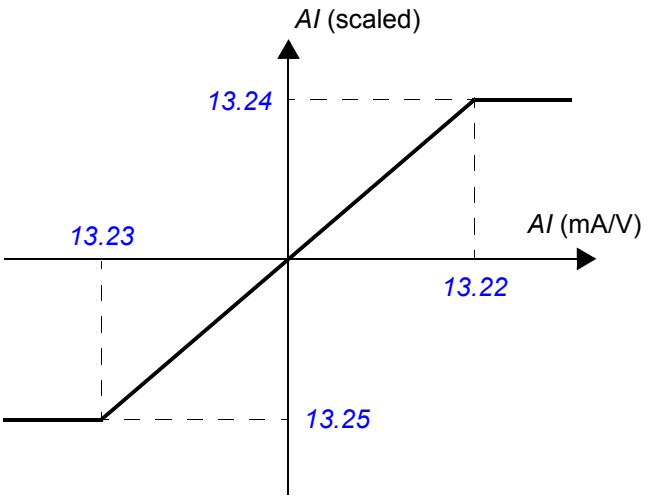
134 Parameters

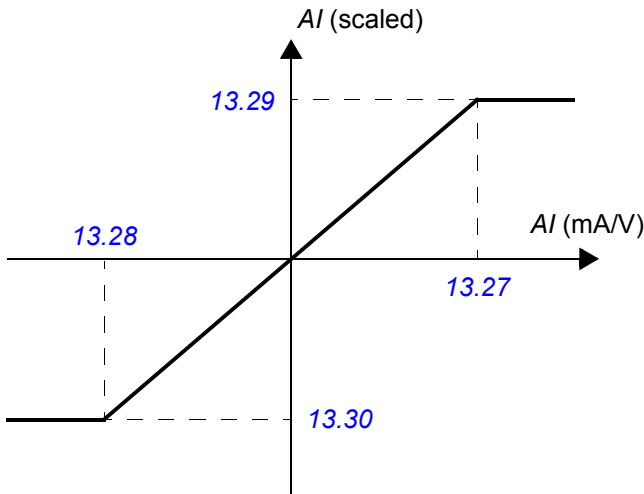
No.	Name/Value	Description	FbEq
13.04	AI1 max scale	<p>Defines the real value that corresponds to the maximum analogue input AI1 value defined by parameter 13.02 AI1 max.</p> 	
	-32768.000 ... 32768.000	Real value corresponding to maximum AI1 value.	1000 = 1
13.05	AI1 min scale	<p>Defines the real value that corresponds to the minimum analogue input AI1 value defined by parameter 13.03 AI1 min. See the drawing at parameter 13.04 AI1 max scale.</p>	
	-32768.000 ...32768.000	Real value corresponding to minimum AI1 value.	1000 = 1
13.06	AI2 filt time	<p>Defines the filter time constant for analogue input AI2. See parameter 13.01 AI1 filt time.</p>	
	0.000 ... 30.000 s	Filter time constant.	1000 = 1 s
13.07	AI2 max	<p>Defines the maximum value for analogue input AI2. The input type is selected with jumper J2 on the JCU Control Unit.</p>	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	AI2 maximum value.	1000 = 1 unit
13.08	AI2 min	<p>Defines the minimum value for analogue input AI2. The input type is selected with jumper J2 on the JCU Control Unit.</p>	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	AI2 minimum value.	1000 = 1 unit



No.	Name/Value	Description	FbEq
13.09	AI2 max scale	Defines the real value that corresponds to the maximum analogue input AI2 value defined by parameter 13.07 AI2 max .	
	-32768.000 ... 32768.000	Real value corresponding to maximum AI2 value.	1000 = 1
13.10	AI2 min scale	Defines the real value that corresponds to the minimum analogue input AI2 value defined by parameter 13.08 AI2 min . See the drawing at parameter 13.09 AI2 max scale .	
	-32768.000 ... 32768.000	Real value corresponding to minimum AI2 value.	1000 = 1
13.11	AI3 filt time	Defines the filter time constant for analogue input AI3. See parameter 13.01 AI1 filt time .	
	0.000 ... 30.000 s	Filter time constant.	1000 = 1 s
13.12	AI3 max	Defines the maximum value for analogue input AI3. The input type depends on the type and/or settings of the I/O extension module installed. See the user documentation of the extension module.	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	AI3 maximum value.	1000 = 1 unit
13.13	AI3 min	Defines the minimum value for analogue input AI3. The input type depends on the type and/or settings of the I/O extension module installed. See the user documentation of the extension module.	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	AI3 minimum value.	1000 = 1 unit

No.	Name/Value	Description	FbEq
13.14	AI3 max scale	<p>Defines the real value that corresponds to the maximum analogue input AI3 value defined by parameter 13.12 AI3 max.</p> 	
	-32768.000 ... 32768.000	Real value corresponding to maximum AI3 value.	1000 = 1
13.15	AI3 min scale	<p>Defines the real value that corresponds to the minimum analogue input AI3 value defined by parameter 13.13 AI3 min. See the drawing at parameter 13.14 AI3 max scale.</p>	
	-32768.000 ... 32768.000	Real value corresponding to minimum AI3 value.	1000 = 1
13.16	AI4 filt time	<p>Defines the filter time constant for analogue input AI4. See parameter 13.01 AI1 filt time.</p>	
	0.000 ... 30.000 s	Filter time constant.	1000 = 1 s
13.17	AI4 max	<p>Defines the maximum value for analogue input AI4. The input type depends on the type and/or settings of the I/O extension module installed. See the user documentation of the extension module.</p>	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	AI4 maximum value.	1000 = 1 unit
13.18	AI4 min	<p>Defines the minimum value for analogue input AI4. The input type depends on the type and/or settings of the I/O extension module installed. See the user documentation of the extension module.</p>	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	AI4 minimum value.	1000 = 1 unit

No.	Name/Value	Description	FbEq
13.19	AI4 max scale	Defines the real value that corresponds to the maximum analogue input AI4 value defined by parameter 13.17 AI4 max .	
	-32768.000 ... 32768.000	Real value corresponding to maximum AI4 value.	1000 = 1
13.20	AI4 min scale	Defines the real value that corresponds to the minimum analogue input AI4 value defined by parameter 13.18 AI4 min . See the drawing at parameter 13.19 AI4 max scale .	
	-32768.000 ... 32768.000	Real value corresponding to minimum AI4 value.	1000 = 1
13.21	AI5 filt time	Defines the filter time constant for analogue input AI5. See parameter 13.01 AI1 filt time .	
	0.000 ... 30.000 s	Filter time constant.	1000 = 1 s
13.22	AI5 max	Defines the maximum value for analogue input AI5. The input type depends on the type and/or settings of the I/O extension module installed. See the user documentation of the extension module.	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	AI5 maximum value.	1000 = 1 unit
13.23	AI5 min	Defines the minimum value for analogue input AI5. The input type depends on the type and/or settings of the I/O extension module installed. See the user documentation of the extension module.	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	AI5 minimum value.	1000 = 1 unit

No.	Name/Value	Description	FbEq
13.24	AI5 max scale	<p>Defines the real value that corresponds to the maximum analogue input AI5 value defined by parameter 13.22 AI5 max.</p> 	
	-32768.000 ... 32768.000	Real value corresponding to maximum AI5 value.	1000 = 1
13.25	AI5 min scale	<p>Defines the real value that corresponds to the minimum analogue input AI5 value defined by parameter 13.23 AI5 min. See the drawing at parameter 13.24 AI5 max scale.</p>	
	-32768.000 ... 32768.000	Real value corresponding to minimum AI5 value.	1000 = 1
13.26	AI6 filt time	<p>Defines the filter time constant for analogue input AI6. See parameter 13.01 AI1 filt time.</p>	
	0.000 ... 30.000 s	Filter time constant.	1000 = 1 s
13.27	AI6 max	<p>Defines the maximum value for analogue input AI6. The input type depends on the type and/or settings of the I/O extension module installed. See the user documentation of the extension module.</p>	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	AI6 maximum value.	1000 = 1 unit
13.28	AI6 min	<p>Defines the minimum value for analogue input AI6. The input type depends on the type and/or settings of the I/O extension module installed. See the user documentation of the extension module.</p>	
	-22.000 ... 22.000 mA or -11.000 ... 11.000 V	AI6 minimum value.	1000 = 1 unit

No.	Name/Value	Description	FbEq
13.29	AI6 max scale	Defines the real value that corresponds to the maximum analogue input AI6 value defined by parameter 13.27 AI6 max . 	
	-32768.000 ... 32768.000	Real value corresponding to maximum AI6 value.	1000 = 1
13.30	AI6 min scale	Defines the real value that corresponds to the minimum analogue input AI6 value defined by parameter 13.28 AI6 min . See the drawing at parameter 13.29 AI6 max scale .	
	-32768.000 ... 32768.000	Real value corresponding to minimum AI6 value.	1000 = 1
13.31	AI tune	Triggers the AI tuning function. Connect the signal to the input and select the appropriate tuning function.	
	No action	AI tune is not activated.	0
	AI1 min tune	Current analogue input AI1 signal value is set as minimum value of AI1 into parameter 13.03 AI1 min . The value reverts back to <i>No action</i> automatically.	1
	AI1 max tune	Current analogue input AI1 signal value is set as maximum value of AI1 into parameter 13.02 AI1 max . The value reverts back to <i>No action</i> automatically.	2
	AI2 min tune	Current analogue input AI2 signal value is set as minimum value of AI2 into parameter 13.08 AI2 min . The value reverts back to <i>No action</i> automatically.	3
	AI2 max tune	Current analogue input AI2 signal value is set as maximum value of AI2 into parameter 13.07 AI2 max . The value reverts back to <i>No action</i> automatically.	4
13.32	AI superv func	Selects how the drive reacts when analogue input signal limit is reached. The limit is selected by parameter 13.33 AI superv cw .	
	No	No action taken.	0
	Fault	The drive trips on an AI SUPERVISION fault.	1

No.	Name/Value	Description	FbEq															
	Spd ref Safe	The drive generates an AI SUPERVISION alarm and sets the speed to the speed defined by parameter 30.02 Speed ref safe .  WARNING! Make sure that it is safe to continue operation in case of a communication break.	2															
	Last speed	The drive generates an AI SUPERVISION alarm and freezes the speed to the level the drive was operating at. The speed is determined by the average speed over the previous 10 seconds.  WARNING! Make sure that it is safe to continue operation in case of a communication break.	3															
13.33	AI superv cw	Selects the analogue input signal supervision limit.																
<table border="1"> <thead> <tr> <th>Bit</th> <th>Supervision</th> <th>Action selected by parameter 13.32 AI superv func is taken if</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>AI1<min</td> <td>AI1 signal value falls below the value defined by equation: par. 13.03 AI1 min - 0.5 mA or V</td> </tr> <tr> <td>1</td> <td>AI1>max</td> <td>AI1 signal value exceeds the value defined by equation: par. 13.02 AI1 max + 0.5 mA or V</td> </tr> <tr> <td>2</td> <td>AI2<min</td> <td>AI2 signal value falls below the value defined by equation: par. 13.08 AI2 min - 0.5 mA or V</td> </tr> <tr> <td>3</td> <td>AI2>max</td> <td>AI2 signal value exceeds the value defined by equation: par. 13.07 AI2 max + 0.5 mA or V</td> </tr> </tbody> </table>				Bit	Supervision	Action selected by parameter 13.32 AI superv func is taken if	0	AI1<min	AI1 signal value falls below the value defined by equation: par. 13.03 AI1 min - 0.5 mA or V	1	AI1>max	AI1 signal value exceeds the value defined by equation: par. 13.02 AI1 max + 0.5 mA or V	2	AI2<min	AI2 signal value falls below the value defined by equation: par. 13.08 AI2 min - 0.5 mA or V	3	AI2>max	AI2 signal value exceeds the value defined by equation: par. 13.07 AI2 max + 0.5 mA or V
Bit	Supervision	Action selected by parameter 13.32 AI superv func is taken if																
0	AI1<min	AI1 signal value falls below the value defined by equation: par. 13.03 AI1 min - 0.5 mA or V																
1	AI1>max	AI1 signal value exceeds the value defined by equation: par. 13.02 AI1 max + 0.5 mA or V																
2	AI2<min	AI2 signal value falls below the value defined by equation: par. 13.08 AI2 min - 0.5 mA or V																
3	AI2>max	AI2 signal value exceeds the value defined by equation: par. 13.07 AI2 max + 0.5 mA or V																
		Example: If parameter value is set to 0b0010, bit 1 AI1>max is selected.																

14 Digital I/O		Configuration of digital input/outputs and relay outputs.																			
14.01	DI invert mask	Inverts status of digital inputs as reported by 02.01 DI status .																			
<table border="1"> <thead> <tr> <th>Bit</th> <th>Name</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1 = Invert DI1</td> </tr> <tr> <td>1</td> <td>1 = Invert DI2</td> </tr> <tr> <td>2</td> <td>1 = Invert DI3</td> </tr> <tr> <td>3</td> <td>1 = Invert DI4</td> </tr> <tr> <td>4</td> <td>1 = Invert DI5</td> </tr> <tr> <td>5</td> <td>1 = Invert DI6</td> </tr> <tr> <td>6</td> <td>Reserved</td> </tr> <tr> <td>7</td> <td>1 = Invert DI8 (on optional FIO-21 I/O Extension)</td> </tr> </tbody> </table>				Bit	Name	0	1 = Invert DI1	1	1 = Invert DI2	2	1 = Invert DI3	3	1 = Invert DI4	4	1 = Invert DI5	5	1 = Invert DI6	6	Reserved	7	1 = Invert DI8 (on optional FIO-21 I/O Extension)
Bit	Name																				
0	1 = Invert DI1																				
1	1 = Invert DI2																				
2	1 = Invert DI3																				
3	1 = Invert DI4																				
4	1 = Invert DI5																				
5	1 = Invert DI6																				
6	Reserved																				
7	1 = Invert DI8 (on optional FIO-21 I/O Extension)																				
14.02	DIO1 conf	Selects whether DIO1 is used as a digital output, digital input or frequency input.																			
	Output	DIO1 is used as a digital output.	0																		
	Input	DIO1 is used as a digital input.	1																		
	Freq input	DIO1 is used as a frequency input.	2																		
14.03	DIO1 out src	Selects a drive signal to be connected to digital output DIO1 (when 14.02 DIO1 conf is set to <i>Output</i>).																			
	Brake cmd	03.16 Brake command (see page 115).	1073742608																		
	Ready	Bit 0 of 06.01 Status word 1 (see page 116).	1073743361																		

No.	Name/Value	Description	FbEq
	Enabled	Bit 1 of <i>06.01 Status word1</i> (see page 116).	1073808897
	Started	Bit 2 of <i>06.01 Status word1</i> (see page 116).	1073874433
	Running	Bit 3 of <i>06.01 Status word1</i> (see page 116).	1073939969
	Alarm	Bit 7 of <i>06.01 Status word1</i> (see page 116).	1074202113
	Ext2 active	Bit 8 of <i>06.01 Status word1</i> (see page 116).	1074267649
	Fault	Bit 10 of <i>06.01 Status word1</i> (see page 116).	1074398721
	Fault(-1)	Bit 12 of <i>06.01 Status word1</i> (see page 116).	1074529793
	Ready relay	Bit 2 of <i>06.02 Status word2</i> (see page 117).	1073874434
	RunningRelay	Bit 3 of <i>06.02 Status word2</i> (see page 117).	1073939970
	Ref running	Bit 4 of <i>06.02 Status word2</i> (see page 117).	1074005506
	Charge ready	Bit 9 of <i>06.02 Status word2</i> (see page 117).	1074333186
	Neg speed	Bit 0 of <i>06.03 Speed ctrl stat</i> (see page 118).	1073743363
	Zero speed	Bit 1 of <i>06.03 Speed ctrl stat</i> (see page 118).	1073808899
	Above limit	Bit 2 of <i>06.03 Speed ctrl stat</i> (see page 118).	1073874435
	At setpoint	Bit 3 of <i>06.03 Speed ctrl stat</i> (see page 118).	1073939971
	Supervision1	Bit 0 of <i>06.13 Superv status</i> (see page 119).	1073743373
	Supervision2	Bit 1 of <i>06.13 Superv status</i> (see page 119).	1073808909
	Supervision3	Bit 2 of <i>06.13 Superv status</i> (see page 119).	1073874445
	Const	Bit pointer setting (see <i>Terms and abbreviations</i> on page 102).	-
	Pointer		
14.04	DIO1 Ton	Defines the on (activation) delay for digital input/output DIO1 when <i>14.02 DIO1 conf</i> is set to <i>Output</i> .	
<p style="text-align: center;"> t_{On} t_{Off} t_{On} t_{Off} </p> <p style="text-align: center;"> t_{On} <i>14.04 DIO1 Ton</i> t_{Off} <i>14.05 DIO1 Toff</i> </p>			
	0.0 ... 3000.0 s	On (activation) delay for DIO1 when set as an output.	10 = 1 s
14.05	DIO1 Toff	Defines the off (deactivation) delay for digital input/output DIO1 when <i>14.02 DIO1 conf</i> is set to <i>Output</i> . See parameter <i>14.04 DIO1 Ton</i> .	
	0.0 ... 3000.0 s	Off (deactivation) delay for DIO1 when set as an output.	10 = 1 s
14.06	DIO2 conf	Selects whether DIO2 is used as a digital output, digital input or frequency output.	
	Output	DIO2 is used as a digital output.	0

No.	Name/Value	Description	FbEq
	Input	DIO2 is used as a digital input.	1
	Freq output	DIO2 is used as a frequency output.	3
14.07	DIO2 out src	Selects a drive signal to be connected to digital output DIO2 (when 14.06 DIO2 conf is set to <i>Output</i>).	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
14.08	DIO2 Ton	Defines the on (activation) delay for digital input/output DIO2 when 14.06 DIO2 conf is set to <i>Output</i> .	
<p style="text-align: center;"> t_{On} 14.08 DIO2 Ton t_{Off} 14.09 DIO2 Toff </p>			
	0.0 ... 3000.0 s	On (activation) delay for DIO2 when set as an output.	10 = 1 s

No.	Name/Value	Description	FbEq
14.09	DIO2 Toff	Defines the off (deactivation) delay for digital input/output DIO2 when 14.06 DIO2 conf is set to <i>Output</i> . See parameter 14.08 DIO2 Ton .	
	0.0 ... 3000.0 s	Off (deactivation) delay for DIO2 when set as an output.	10 = 1 s
14.10	DIO3 conf	Selects whether DIO3 is used as a digital output or input.	
	Output	DIO3 is used as a digital output.	0
	Input	DIO3 is used as a digital input.	1
14.11	DIO3 out src	Selects a drive signal to be connected to digital output DIO3 (when 14.10 DIO3 conf is set to <i>Output</i>).	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
14.14	DIO4 conf	Selects whether DIO4 is used as a digital output or input.	
	Output	DIO4 is used as a digital output.	0
	Input	DIO4 is used as a digital input.	1
14.15	DIO4 out src	Selects a drive signal to be connected to digital output DIO4 (when 14.14 DIO4 conf is set to <i>Output</i>).	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113

144 Parameters

No.	Name/Value	Description	FbEq
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
14.18	DIO5 conf	Selects whether DIO5 is used as a digital output or input.	
	Output	DIO5 is used as a digital output.	0
	Input	DIO5 is used as a digital input.	1
14.19	DIO5 out src	Selects a drive signal to be connected to digital output DIO5 (when 14.18 DIO5 conf is set to Output).	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445

No.	Name/Value	Description	FbEq
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
14.22	DIO6 conf	Selects whether DIO6 is used as a digital output or input.	
	Output	DIO6 is used as a digital output.	0
	Input	DIO6 is used as a digital input.	1
14.23	DIO6 out src	Selects a drive signal to be connected to digital output DIO6 (when 14.22 DIO6 conf is set to Output).	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
14.26	DIO7 conf	Selects whether DIO7 is used as a digital output or input.	
	Output	DIO7 is used as a digital output.	0
	Input	DIO7 is used as a digital input.	1
14.27	DIO7 out src	Selects a drive signal to be connected to digital output DIO7 (when 14.26 DIO7 conf is set to Output).	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721

No.	Name/Value	Description	FbEq
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
14.30	DIO8 conf	Selects whether DIO8 is used as a digital output or input.	
	Output	DIO8 is used as a digital output.	0
	Input	DIO8 is used as a digital input.	1
14.31	DIO8 out src	Selects a drive signal to be connected to digital output DIO8 (when 14.30 DIO8 conf is set to Output).	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		

No.	Name/Value	Description	FbEq
14.34	DIO9 conf	Selects whether DIO9 is used as a digital output or input.	
	Output	DIO9 is used as a digital output.	0
	Input	DIO9 is used as a digital input.	1
14.35	DIO9 out src	Selects a drive signal to be connected to digital output DIO9 (when 14.34 DIO9 conf is set to <i>Output</i>).	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
14.38	DIO10 conf	Selects whether DIO10 is used as a digital output or input.	
	Output	DIO10 is used as a digital output.	0
	Input	DIO10 is used as a digital input.	1
14.39	DIO10 out src	Selects a drive signal to be connected to digital output DIO10 (when 14.38 DIO10 conf is set to <i>Output</i>).	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434

No.	Name/Value	Description	FbEq
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
14.42	RO1 src	Selects a drive signal to be connected to relay output RO1.	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		

No.	Name/Value	Description	FbEq
14.43	RO1 Ton	Defines the on (activation) delay for relay output RO1.	
<p style="text-align: center;"> t_{On} 14.43 RO1 Ton t_{Off} 14.44 RO1 Toff </p>			
	0.0 ... 3000.0 s	On (activation) delay for RO1.	10 = 1 s
14.44	RO1 Toff	Defines the off (deactivation) delay for relay output RO1. See parameter 14.43 RO1 Ton .	
	0.0 ... 3000.0 s	Off (deactivation) delay for RO1.	10 = 1 s
14.45	RO2 src	Selects a drive signal to be connected to relay output RO2.	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		

150 Parameters

No.	Name/Value	Description	FbEq
14.48	RO3 src	Selects a drive signal to be connected to relay output RO3.	
	Brake cmd	03.16 Brake command (see page 115).	1073742608
	Ready	Bit 0 of 06.01 Status word1 (see page 116).	1073743361
	Enabled	Bit 1 of 06.01 Status word1 (see page 116).	1073808897
	Started	Bit 2 of 06.01 Status word1 (see page 116).	1073874433
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969
	Alarm	Bit 7 of 06.01 Status word1 (see page 116).	1074202113
	Ext2 active	Bit 8 of 06.01 Status word1 (see page 116).	1074267649
	Fault	Bit 10 of 06.01 Status word1 (see page 116).	1074398721
	Fault(-1)	Bit 12 of 06.01 Status word1 (see page 116).	1074529793
	Ready relay	Bit 2 of 06.02 Status word2 (see page 117).	1073874434
	RunningRelay	Bit 3 of 06.02 Status word2 (see page 117).	1073939970
	Ref running	Bit 4 of 06.02 Status word2 (see page 117).	1074005506
	Charge ready	Bit 9 of 06.02 Status word2 (see page 117).	1074333186
	Neg speed	Bit 0 of 06.03 Speed ctrl stat (see page 118).	1073743363
	Zero speed	Bit 1 of 06.03 Speed ctrl stat (see page 118).	1073808899
	Above limit	Bit 2 of 06.03 Speed ctrl stat (see page 118).	1073874435
	At setpoint	Bit 3 of 06.03 Speed ctrl stat (see page 118).	1073939971
	Supervision1	Bit 0 of 06.13 Superv status (see page 119).	1073743373
	Supervision2	Bit 1 of 06.13 Superv status (see page 119).	1073808909
	Supervision3	Bit 2 of 06.13 Superv status (see page 119).	1073874445
	Const Pointer	Bit pointer setting (see Terms and abbreviations on page 102).	-
14.51	RO4 src	Selects a drive signal to be connected to relay output RO4.	
	Const Pointer	Bit pointer setting (see Terms and abbreviations on page 102).	-
14.54	RO5 src	Selects a drive signal to be connected to relay output RO5.	
	Const Pointer	Bit pointer setting (see Terms and abbreviations on page 102).	-

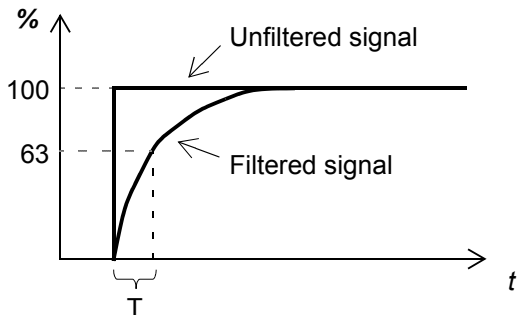
No.	Name/Value	Description	FbEq
14.57	Freq in max	<p>Defines the maximum input frequency for DIO1 when parameter 14.02 DIO1 conf is set to <i>Freq input</i>. The frequency signal connected to DIO1 is scaled into an internal signal (02.20 Freq in) by parameters 14.57...14.60 as follows:</p>	
	3 ... 32768 Hz	DIO1 maximum frequency.	1 = 1 Hz
14.58	Freq in min	Defines the minimum input frequency for DIO1 when parameter 14.02 DIO1 conf is set to <i>Freq input</i> . See parameter 14.57 Freq in max .	
	3 ... 32768 Hz	DIO1 minimum frequency.	1 = 1 Hz
14.59	Freq in max scal	Defines the value that corresponds to the maximum input frequency defined by parameter 14.57 Freq in max . See parameter 14.57 Freq in max .	
	-32768 ... 32768	Scaled value corresponding to DIO1 maximum frequency.	1 = 1
14.60	Freq in min scal	Defines the value that corresponds to the minimum input frequency defined by parameter 14.58 Freq in min . See parameter 14.57 Freq in max .	
	-32768 ... 32768	Scaled value corresponding to DIO1 minimum frequency.	1 = 1
14.61	Freq out src	Selects a drive signal to be connected to frequency output DIO2 (when 14.06 DIO2 conf is set to <i>Freq output</i>).	
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-

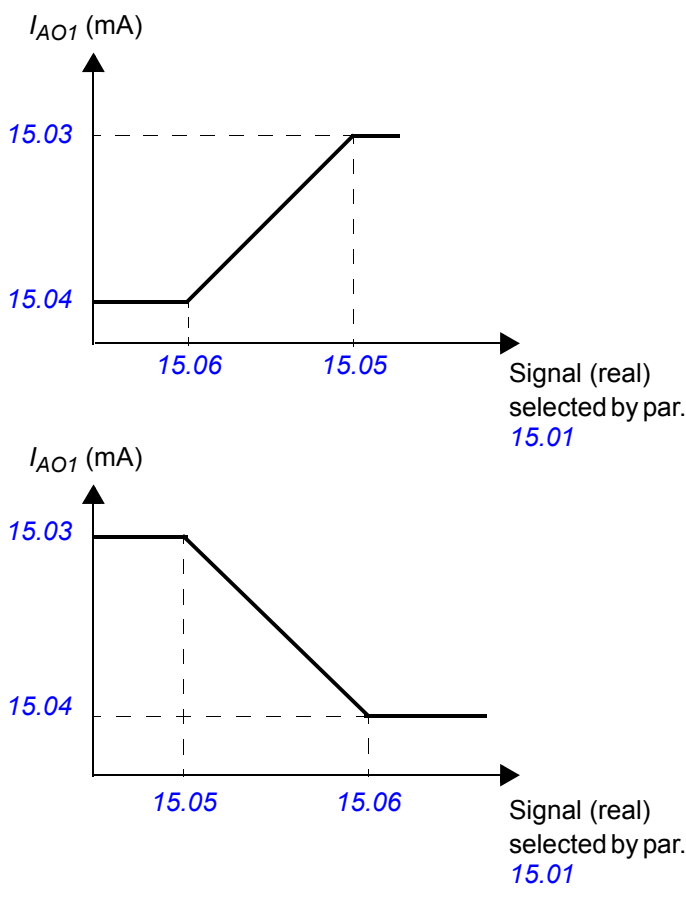
No.	Name/Value	Description	FbEq
14.62	Freq out max src	<p>When 14.06 DIO2 conf is set to <i>Freq output</i>, defines the real value of the signal (selected by parameter 14.61 Freq out src) that corresponds to the maximum DIO2 frequency output value (defined by parameter 14.64 Freq out max sca).</p> <p>The figure contains two graphs. The top graph plots f_{DIO2} (Hz) on the y-axis against 'Signal (real) selected by par. 14.61' on the x-axis. The y-axis has values 14.64 and 14.65. The x-axis has values 14.63 and 14.62. The curve is constant at 14.65 Hz for signal values up to 14.63, then rises linearly to 14.64 Hz at signal value 14.62, and remains constant at 14.64 Hz for higher signal values. The bottom graph plots f_{DIO2} (Hz) on the y-axis against 'Signal (real) selected by par. 14.61' on the x-axis. The y-axis has values 14.64 and 14.65. The x-axis has values 14.62 and 14.63. The curve is constant at 14.64 Hz for signal values up to 14.62, then falls linearly to 14.65 Hz at signal value 14.63, and remains constant at 14.65 Hz for higher signal values.</p>	
	0 ... 32768	Real signal value corresponding to maximum DIO2 output frequency.	1 = 1
14.63	Freq out min src	When 14.06 DIO2 conf is set to <i>Freq output</i> , defines the real value of the signal (selected by parameter 14.61 Freq out src) that corresponds to the minimum DIO2 frequency output value (defined by parameter 14.65 Freq out min sca).	
	0 ... 32768	Real signal value corresponding to minimum DIO2 output frequency.	1 = 1
14.64	Freq out max sca	When 14.06 DIO2 conf is set to <i>Freq output</i> , defines the maximum DIO2 output frequency.	
	3 ... 32768 Hz	Maximum DIO2 output frequency.	1 = 1 Hz
14.65	Freq out min sca	When 14.06 DIO2 conf is set to <i>Freq output</i> , defines the minimum DIO2 output frequency.	
	3 ... 32768 Hz	Minimum DIO2 output frequency.	1 = 1 Hz
14.66	RO6 src	Selects a drive signal to be connected to relay output RO6.	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
14.69	RO7 src	Selects a drive signal to be connected to relay output RO7.	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		

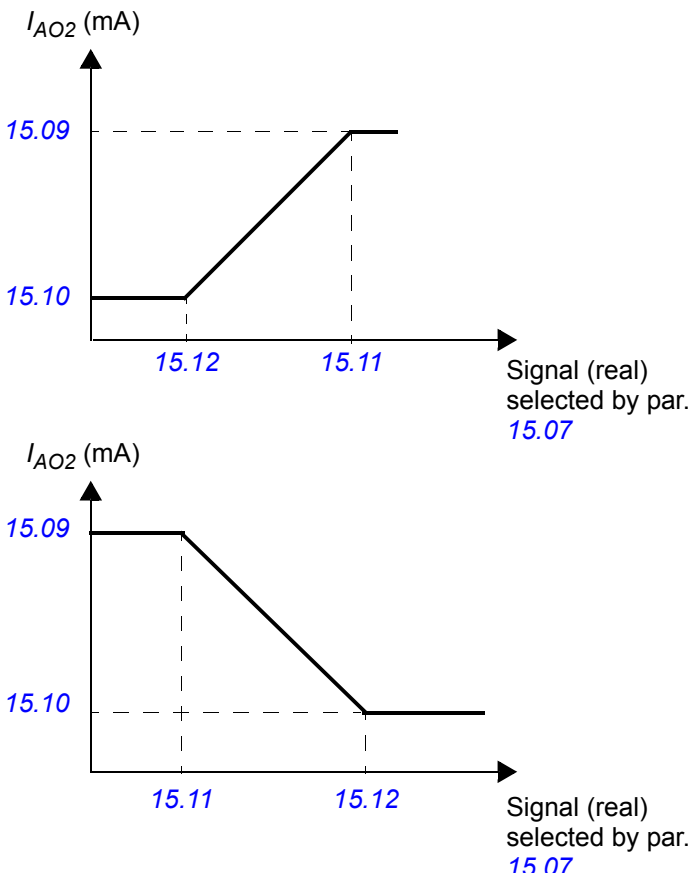
No.	Name/Value	Description	FbEq
14.72	DIO invert mask	Inverts status of digital input/outputs as reported by 02.03 DIO status .	
	Bit	Name	
	0	1 = Invert DIO1	
	1	1 = Invert DIO2	
	2	1 = Invert DIO3 (on optional FIO-01 I/O Extension)	
	3	1 = Invert DIO4 (on optional FIO-01 I/O Extension)	
	4	1 = Invert DIO5 (on optional FIO-01 I/O Extension)	
	5	1 = Invert DIO6 (on optional FIO-01 I/O Extension)	
	6	1 = Invert DIO7 (on optional FIO-01 I/O Extension)	
	7	1 = Invert DIO8 (on optional FIO-01 I/O Extension)	
	8	1 = Invert DIO9 (on optional FIO-01 I/O Extension)	
	9	1 = Invert DIO10 (on optional FIO-01 I/O Extension)	

15 Analogue outputs		Selection and processing of actual signals to be indicated through the analogue outputs. See section Programmable analog outputs on page 59.	
15.01	AO1 src	Selects a drive signal to be connected to analogue output AO1.	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	SpRef unramp	03.03 SpeedRef unramp (see page 114).	1073742595
	SpRef ramped	03.05 SpeedRef ramped (see page 114).	1073742597
	SpRef used	03.06 SpeedRef used (see page 114).	1073742598
	TorqRef used	03.14 Torq ref used (see page 115).	1073742606
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-

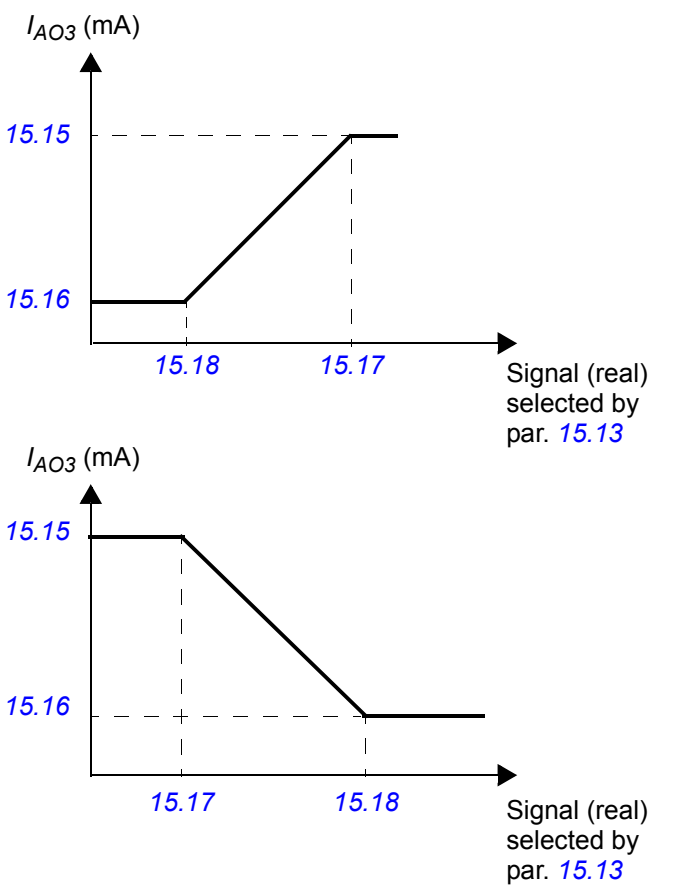
154 Parameters

No.	Name/Value	Description	FbEq
15.02	AO1 filt time	Defines the filtering time constant for analogue output AO1.  $O = I \times (1 - e^{-t/T})$ I = filter input (step) O = filter output t = time T = filter time constant	
	0.000 ... 30.000 s	Filter time constant.	1000 = 1 s
15.03	AO1 out max	Defines the maximum output value for analogue output AO1.	
	0.000 ... 22.700 mA	Maximum AO1 output value.	1000 = 1 mA
15.04	AO1 out min	Defines the minimum output value for analogue output AO1.	
	0.000 ... 22.700 mA	Minimum AO1 output value.	1000 = 1 mA

No.	Name/Value	Description	FbEq
15.05	AO1 src max	<p>Defines the real value of the signal (selected by parameter 15.01 AO1 src) that corresponds to the maximum AO1 output value (defined by parameter 15.03 AO1 out max).</p> 	
	-32768.000 ... 32768.000	Real signal value corresponding to maximum AO1 output value.	1000 = 1
15.06	AO1 src min	<p>Defines the real value of the signal (selected by parameter 15.01 AO1 src) that corresponds to the minimum AO1 output value (defined by parameter 15.04 AO1 out min). See parameter 15.05 AO1 src max.</p>	
	-32768.000 ... 32768.000	Real signal value corresponding to minimum AO1 output value.	1000 = 1
15.07	AO2 src	Selects a drive signal to be connected to analogue output AO2.	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	SpRef unramp	03.03 SpeedRef unramp (see page 114).	1073742595


No.	Name/Value	Description	FbEq
	SpRef ramped	03.05 SpeedRef ramped (see page 114).	1073742597
	SpRef used	03.06 SpeedRef used (see page 114).	1073742598
	TorqRef used	03.14 Torq ref used (see page 115).	1073742606
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
15.08	AO2 filt time	Defines the filtering time constant for analogue output AO2. See parameter 15.02 AO1 filt time .	
	0.000 ... 30.000 s	Filter time constant.	1000 = 1 s
15.09	AO2 out max	Defines the maximum output value for analogue output AO2.	
	0.000 ... 22.700 mA	Maximum AO2 output value.	1000 = 1 mA
15.10	AO2 out min	Defines the minimum output value for analogue output AO2.	
	0.000 ... 22.700 mA	Minimum AO2 output value.	1000 = 1 mA
15.11	AO2 src max	<p>Defines the real value of the signal (selected by parameter 15.07 AO2 src) that corresponds to the maximum AO2 output value (defined by parameter 15.09 AO2 out max).</p> 	
	-32768.000 ... 32768.000	Real signal value corresponding to maximum AO2 output value.	1000 = 1

No.	Name/Value	Description	FbEq
15.12	AO2 src min	Defines the real value of the signal (selected by parameter 15.07 AO2 src) that corresponds to the minimum AO1 output value (defined by parameter 15.10 AO2 out min). See parameter 15.11 AO2 src max .	
	-32768.000 ... 32768.000	Real signal value corresponding to minimum AO2 output value.	1000 = 1
15.13	AO3 src	Selects a drive signal to be connected to analogue output AO3.	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	SpRef unramp	03.03 SpeedRef unramp (see page 114).	1073742595
	SpRef ramped	03.05 SpeedRef ramped (see page 114).	1073742597
	SpRef used	03.06 SpeedRef used (see page 114).	1073742598
	TorqRef used	03.14 Torq ref used (see page 115).	1073742606
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
15.14	AO3 filt time	Defines the filtering time constant for analogue output AO3. See parameter 15.02 AO1 filt time .	
	0.000 ... 30.000 s	Filter time constant.	1000 = 1 s
15.15	AO3 out max	Defines the maximum output value for analogue output AO3.	
	0.000 ... 22.700 mA	Maximum AO3 output value.	1000 = 1 mA
15.16	AO3 out min	Defines the minimum output value for analogue output AO3.	
	0.000 ... 22.700 mA	Minimum AO3 output value.	1000 = 1 mA

No.	Name/Value	Description	FbEq
15.17	AO3 src max	<p>Defines the real value of the signal (selected by parameter 15.13 AO3 src) that corresponds to the maximum AO3 output value (defined by parameter 15.15 AO3 out max).</p>  <p>The figure contains two graphs. The top graph plots I_{AO3} (mA) on the y-axis against 'Signal (real) selected by par. 15.13' on the x-axis. The y-axis has values 15.15 and 15.16. The x-axis has values 15.18 and 15.17. The curve is constant at 15.16 mA for signals up to 15.18, then rises linearly to 15.15 mA at signal 15.17, and remains constant thereafter. The bottom graph plots I_{AO3} (mA) on the y-axis against 'Signal (real) selected by par. 15.13' on the x-axis. The y-axis has values 15.15 and 15.16. The x-axis has values 15.17 and 15.18. The curve is constant at 15.15 mA for signals up to 15.17, then falls linearly to 15.16 mA at signal 15.18, and remains constant thereafter.</p>	
	-32768.000 ... 32768.000	Real signal value corresponding to maximum AO3 output value.	1000 = 1
15.18	AO3 src min	Defines the real value of the signal (selected by parameter 15.13 AO3 src) that corresponds to the minimum AO3 output value (defined by parameter 15.16 AO3 out min). See parameter 15.17 AO3 src max .	
	-32768.000 ... 32768.000	Real signal value corresponding to minimum AO3 output value.	1000 = 1
15.19	AO4 src	Selects a drive signal to be connected to analogue output AO4.	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	SpRef unramp	03.03 SpeedRef unramp (see page 114).	1073742595

No.	Name/Value	Description	FbEq
	SpRef ramped	03.05 SpeedRef ramped (see page 114).	1073742597
	SpRef used	03.06 SpeedRef used (see page 114).	1073742598
	TorqRef used	03.14 Torq ref used (see page 115).	1073742606
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
15.20	AO4 filt time	Defines the filtering time constant for analogue output AO4. See parameter 15.02 AO1 filt time .	
	0.000 ... 30.000 s	Filter time constant.	1000 = 1 s
15.21	AO4 out max	Defines the maximum output value for analogue output AO4.	
	0.000 ... 22.700 mA	Maximum AO4 output value.	1000 = 1 mA
15.22	AO4 out min	Defines the minimum output value for analogue output AO4.	
	0.000 ... 22.700 mA	Minimum AO4 output value.	1000 = 1 mA
15.23	AO4 src max	Defines the real value of the signal (selected by parameter 15.19 AO4 src) that corresponds to the maximum AO4 output value (defined by parameter 15.21 AO4 out max).	
		<p>The figure contains two graphs. Both graphs have I_{AO4} (mA) on the vertical axis and 'Signal (real) selected by par. 15.19' on the horizontal axis. The top graph shows a signal value increasing from 15.22 to 15.23. The corresponding I_{AO4} value starts at 15.22 mA, remains constant until the signal reaches 15.24, then increases linearly to 15.21 mA at signal 15.23, and remains constant thereafter. The bottom graph shows a signal value increasing from 15.23 to 15.24. The corresponding I_{AO4} value starts at 15.21 mA, remains constant until the signal reaches 15.23, then decreases linearly to 15.22 mA at signal 15.24, and remains constant thereafter.</p>	
	-32768.000 ... 32768.000	Real signal value corresponding to maximum AO4 output value.	1000 = 1

No.	Name/Value	Description	FbEq											
15.24	AO4 src min	Defines the real value of the signal (selected by parameter 15.19 AO4 src) that corresponds to the minimum AO4 output value (defined by parameter 15.22 AO4 out min). See parameter 15.23 AO4 src max .												
	-32768.000 ... 32768.000	Real signal value corresponding to minimum AO4 output value.	1000 = 1											
15.25	AO ctrl word	Defines how a signed source is processed before output.												
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Name</th> <th>Information</th> </tr> </thead> <tbody> <tr> <td rowspan="2">0</td> <td rowspan="2">AO1 func</td> <td>1 = AO1 is bipolar</td> </tr> <tr> <td>0 = AO1 is absolute value of source</td> </tr> <tr> <td rowspan="2">1</td> <td rowspan="2">AO2 func</td> <td>1 = AO2 is bipolar</td> </tr> <tr> <td>0 = AO2 is absolute value of source</td> </tr> </tbody> </table>	Bit	Name	Information	0	AO1 func	1 = AO1 is bipolar	0 = AO1 is absolute value of source	1	AO2 func	1 = AO2 is bipolar	0 = AO2 is absolute value of source	
Bit	Name	Information												
0	AO1 func	1 = AO1 is bipolar												
		0 = AO1 is absolute value of source												
1	AO2 func	1 = AO2 is bipolar												
		0 = AO2 is absolute value of source												

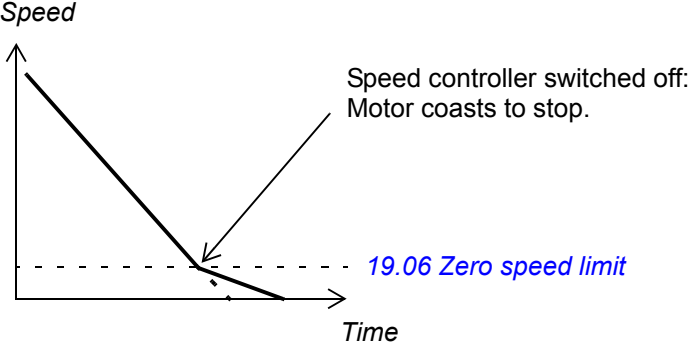
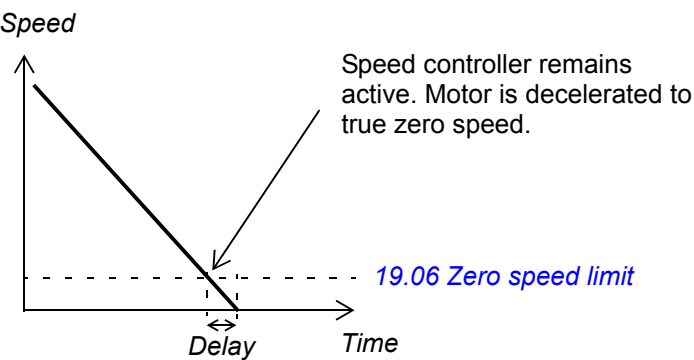
16 System		Parameter lock, parameter restore, user parameter sets etc.	
16.01	Local lock	<p>Selects the source for disabling local control (Take/Release button in the PC tool, LOC/REM key of the panel).</p> <p>0 = Local control enabled.</p> <p>1 = Local control disabled.</p> <p> WARNING! Before activating, ensure that the control panel is not needed for stopping the drive!</p>	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
16.02	Parameter lock	Selects the state of the parameter lock. The lock prevents parameter changing.	
	Locked	Locked. Parameter values cannot be changed from the control panel. The lock can be opened by entering the valid code into parameter 16.03 Pass code .	0
	Open	The lock is open. Parameter values can be changed.	1
	Not saved	The lock is open. Parameter values can be changed, but the changes will not be stored at power switch-off.	2
16.03	Pass code	<p>Selects the pass code for the parameter lock (see parameter 16.02 Parameter lock).</p> <p>After entering 358 at this parameter, parameter 16.02 Parameter lock can be adjusted. The value reverts back to 0 automatically.</p>	
	0 ... 2147483647	Pass code for parameter lock.	1 = 1
16.04	Param restore	<p>Restores the original settings of the application, i.e. parameter factory default values.</p> <p>Note: This parameter cannot be changed while the drive is running.</p>	
	Done	Restoring is completed.	0
	Restore defs	All parameter values are restored to default values, except motor data, ID run results, and fieldbus adapter, drive-to-drive link and encoder configuration data.	1
	Clear all	All parameter values are restored to default values, including motor data, ID run results and fieldbus adapter and encoder configuration data. PC tool communication is interrupted during the restoring. Drive CPU is re-booted after the restoring is completed.	2

No.	Name/Value	Description	FbEq
16.07	Param save	Saves the valid parameter values to the permanent memory. Note: A new parameter value is saved automatically when changed from the PC tool or panel but not when altered through a fieldbus adapter connection.	
	Done	Save completed.	0
	Save	Save in progress.	1
16.09	User set sel	Enables the saving and restoring of up to four custom sets of parameter settings. The set that was in use before powering down the drive is in use after the next power-up. Notes: <ul style="list-style-type: none"> Fieldbus adapter and encoder parameters (groups 50-53 and 90-93 respectively) are not part of the user parameter sets. Any parameter changes made after loading a set are not automatically stored – they must be saved using this parameter. 	
	No request	Load or save operation complete; normal operation.	1
	Load set 1	Load user parameter set 1.	2
	Load set 2	Load user parameter set 2.	3
	Load set 3	Load user parameter set 3.	4
	Load set 4	Load user parameter set 4.	5
	Save set 1	Save user parameter set 1.	6
	Save set 2	Save user parameter set 2.	7
	Save set 3	Save user parameter set 3.	8
	Save set 4	Save user parameter set 4.	9
	IO mode	Load user parameter set using parameters 16.11 User IO sel lo and 16.12 User IO sel hi .	10
16.10	User set log	Shows the status of the user parameter sets (see parameter 16.09 User set sel). Read-only.	
	N/A	No user sets have been saved.	0
	Loading	A user set is being loaded.	1
	Saving	A user set is being saved.	2
	Faulted	Invalid or empty parameter set.	4
	Set1 IO act	User parameter set 1 has been selected by parameters 16.11 User IO sel lo and 16.12 User IO sel hi .	8
	Set2 IO act	User parameter set 2 has been selected by parameters 16.11 User IO sel lo and 16.12 User IO sel hi .	16
	Set3 IO act	User parameter set 3 has been selected by parameters 16.11 User IO sel lo and 16.12 User IO sel hi .	32
	Set4 IO act	User parameter set 4 has been selected by parameters 16.11 User IO sel lo and 16.12 User IO sel hi .	64
	Set1 par act	User parameter set 1 has been loaded using parameter 16.09 User set sel .	128
	Set2 par act	User parameter set 2 has been loaded using parameter 16.09 User set sel .	256
	Set3 par act	User parameter set 3 has been loaded using parameter 16.09 User set sel .	512

162 Parameters

No.	Name/Value	Description	FbEq															
	Set4 par act	User parameter set 4 has been loaded using parameter 16.09 User set sel .	1024															
16.11	User IO sel lo	When parameter 16.09 User set sel is set to <i>IO mode</i> , selects the user parameter set together with parameter 16.12 User IO sel hi . The status of the source defined by this parameter and parameter 16.12 select the user parameter set as follows: <table border="1" data-bbox="475 454 1185 759"> <thead> <tr> <th>Status of source defined by par. 16.11</th> <th>Status of source defined by par. 16.12</th> <th>User parameter set selected</th> </tr> </thead> <tbody> <tr> <td>FALSE</td> <td>FALSE</td> <td>Set 1</td> </tr> <tr> <td>TRUE</td> <td>FALSE</td> <td>Set 2</td> </tr> <tr> <td>FALSE</td> <td>TRUE</td> <td>Set 3</td> </tr> <tr> <td>TRUE</td> <td>TRUE</td> <td>Set 4</td> </tr> </tbody> </table>	Status of source defined by par. 16.11	Status of source defined by par. 16.12	User parameter set selected	FALSE	FALSE	Set 1	TRUE	FALSE	Set 2	FALSE	TRUE	Set 3	TRUE	TRUE	Set 4	
Status of source defined by par. 16.11	Status of source defined by par. 16.12	User parameter set selected																
FALSE	FALSE	Set 1																
TRUE	FALSE	Set 2																
FALSE	TRUE	Set 3																
TRUE	TRUE	Set 4																
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-															
	Pointer																	
16.12	User IO sel hi	See parameter 16.11 User IO sel lo .																
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-															
	Pointer																	
16.14	Reset ChgParLog	Resets the log of latest parameter changes.																
	Done	Reset not requested (normal operation).	0															
	Reset	Reset log of latest parameter changes. The value reverts automatically to <i>Done</i> .	1															
16.15	Menu set sel	Loads the short or long parameter list. By default, the short (selective) parameter list is displayed by drive.																
	No request	No change has been requested.	0															
	Load short	Load short parameter list. Only a selective list of parameters will be displayed.	1															
	Load long	Load long parameter list. All parameters will be displayed.	2															
16.16	Menu set active	Shows which parameter list is active. See parameter 16.15 Menu set sel .																
	None	No parameter list is active.	0															
	Short menu	Short parameter list is active.	1															
	Long menu	Long parameter list is active. All parameters are displayed.	2															
16.17	Power unit	Selects the unit of power for parameters such as 01.22 Power inu out , 01.23 Motor power and 99.10 Mot nom power .																
	kW	Kilowatt.	0															
	hp	Horsepower.	1															
19 Speed calculation		Speed feedback, speed window, etc. settings.																
19.01	Speed scaling	Defines the terminal speed value used in acceleration and the initial speed value used in deceleration (see parameter group 22 Speed ref ramp). Also defines the rpm value that corresponds to 20000 for fieldbus communication with ABB Drives communication profile.																
	0 ... 30000 rpm	Acceleration/deceleration terminal/initial speed.	1 = 1 rpm															

No.	Name/Value	Description	FbEq
19.02	Speed fb sel	Selects the speed feedback value used in control.	
	Estimated	A calculated speed estimate is used.	0
	Enc1 speed	Actual speed measured with encoder 1. The encoder is selected by parameter 90.01 Encoder 1 sel.	1
	Enc2 speed	Actual speed measured with encoder 2. The encoder is selected by parameter 90.02 Encoder 2 sel.	2
19.03	MotorSpeed filt	<p>Defines the time constant of the actual speed filter, i.e. time within the actual speed has reached 63% of the nominal speed (filtered speed = 01.01 Motor speed rpm).</p> <p>If the used speed reference remains constant, the possible interferences in the speed measurement can be filtered with the actual speed filter. Reducing the ripple with filter may cause speed controller tuning problems. A long filter time constant and fast acceleration time contradict one another. A very long filter time results in unstable control.</p> <p>If there are substantial interferences in the speed measurement, the filter time constant should be proportional to the total inertia of the load and motor, in this case 10...30% of the mechanical time constant</p> $t_{\text{mech}} = (n_{\text{nom}} / T_{\text{nom}}) \times J_{\text{tot}} \times 2\pi / 60$, where J_{tot} = total inertia of the load and motor (the gear ratio between the load and motor must be taken into account) n_{nom} = motor nominal speed T_{nom} = motor nominal torque See also parameter 23.07 Speed err Ftime .	
	0.000 ... 10000.000 ms	Time constant of the actual speed filter.	1000 = 1 ms
19.06	Zero speed limit	Defines the zero speed limit. The motor is stopped along a speed ramp until the defined zero speed limit is reached. After the limit, the motor coasts to stop.	
	0.00 ... 30000.00 rpm	Zero speed limit.	100 = 1 rpm

No.	Name/Value	Description	FbEq
19.07	Zero speed delay	<p>Defines the delay for the zero speed delay function. The function is useful in applications where a smooth and quick restarting is essential. During the delay, the drive knows accurately the rotor position.</p> <p>Without Zero Speed Delay: The drive receives a stop command and decelerates along a ramp. When the motor actual speed falls below the value of parameter 19.06 Zero speed limit, the speed controller is switched off. The inverter modulation is stopped and the motor coasts to standstill.</p>  <p>With Zero Speed Delay: The drive receives a stop command and decelerates along a ramp. When the actual motor speed falls below the value of parameter 19.06 Zero speed limit, the zero speed delay function activates. During the delay the function keeps the speed controller live: the inverter modulates, motor is magnetized and the drive is ready for a quick restart. Zero speed delay can be used e.g. with the jogging function.</p> 	
	0 ... 30000 ms	Zero speed delay.	1 = 1 ms
19.08	Above speed lim	Defines the supervision limit for the actual speed. See also parameter 02.13 FBA main sw , bit 10.	
	0 ... 30000 rpm	Actual speed supervision limit.	1 = 1 rpm

No.	Name/Value	Description	FbEq
19.09	Speed TripMargin	<p>Defines, together with 20.01 Maximum speed and 20.02 Minimum speed, the maximum allowed speed of the motor (overspeed protection). If actual speed (01.01 Motor speed rpm) exceeds the speed limit defined by parameter 20.01 or 20.02 by more than the value of this parameter, the drive trips on the OVERSPEED fault.</p> <p>Example: If the maximum speed is 1420 rpm and speed trip margin is 300 rpm, the drive trips at 1720 rpm.</p>	
	0.0 ... 10000.0 rpm	Overspeed trip margin.	10 = 1 rpm
19.10	Speed window	<p>Defines the absolute value for the motor speed window supervision, i.e. the absolute value for the difference between the actual speed and the unramped speed reference (01.01 Motor speed rpm - 03.03 SpeedRef unrampr). When the motor speed is within the limits defined by this parameter, signal 02.24 FBA main sw bit 8 (AT_SETPOINT) is 1. If the motor speed is not within the defined limits, bit 8 is 0.</p>	
	0 ... 30000 rpm	Absolute value for motor speed window supervision.	1 = 1 rpm
19.13	Speed fbk fault	<p>Selects the action in case of speed feedback data loss.</p> <p>Note: If this parameter is set to <i>Warning</i> or <i>No</i>, a loss of feedback will cause an internal faulted state. To clear the internal fault and to reactivate speed feedback, use parameter 90.10 Enc par refresh.</p>	
	Fault	Drive trips on a fault (OPTION COMM LOSS, ENCODER 1/2, ENCODER 1/2 CABLE or SPEED FEEDBACK depending on the type of problem).	0
	Warning	Drive continues operation with open loop control and generates an alarm (OPTION COMM LOSS, ENCODER 1/2 FAILURE, ENCODER 1/2 CABLE or SPEED FEEDBACK depending on the type of the problem).	1
	No	Drive continues operation with open loop control. No faults or alarms are generated.	2

No.	Name/Value	Description	FbEq
20 Limits		Drive operation limits. See also section Speed controller tuning on page 61.	
20.01	Maximum speed	Defines the allowed maximum speed. For safety reasons, after ID run this parameter is set to a 1.2 times bigger value than the nominal motor speed (parameter 99.09 Mot nom speed).	
	0 ... 30000 rpm	Maximum speed.	1 = 1 rpm
20.02	Minimum speed	Defines the allowed minimum speed. For safety reasons, after ID run this parameter is set to a 1.2 times bigger value than the nominal motor speed (parameter 99.09 Mot nom speed).	
	-30000 ... 0 rpm	Minimum speed.	1 = 1 rpm
20.03	Pos speed ena	<p>Selects the source of the positive speed reference enable command.</p> <p>1 = Positive speed reference is enabled. 0 = Positive speed reference is interpreted as zero speed reference (In the figure below 03.03 SpeedRef unramp is set to zero after the positive speed enable signal has cleared).</p> <p>Actions in different control modes:</p> <p>Speed control: Speed reference is set to zero and the motor is stopped along the currently active deceleration ramp.</p> <p>Torque control: Torque limit is set to zero and the rush controller stops the motor.</p>	
		<p>The diagram illustrates the motor's response to a change in speed enable signals. It shows four signals over time: 20.03 Pos speed ena (a pulse that drops to zero), 20.04 Neg speed ena (a pulse that rises to one), 03.03 SpeedRef unramp (a signal that drops to zero when 20.03 Pos speed ena drops), and 01.08 Encoder1 speed (which ramps down to zero as 03.03 SpeedRef unramp drops to zero, and then ramps up again when 03.03 SpeedRef unramp rises back to its original level).</p>	
		<p>Example: The motor is rotating in the forward direction. To stop the motor, the positive speed enable signal is deactivated by a hardware limit switch (e.g. via digital input). If the positive speed enable signal remains deactivated and the negative speed enable signal is active, only reverse rotation of the motor is allowed.</p>	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
20.04	Neg speed ena	Selects the source of the negative speed reference enable command. See parameter 20.03 Pos speed ena .	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
20.05	Maximum current	Defines the maximum allowed motor current.	
	0.00 ... 30000.00 A	Maximum motor current.	100 = 1 A

No.	Name/Value	Description	FbEq
20.06	Torq lim sel	Defines a source that selects between the two sets of torque limits defined by parameters 20.07...20.10 . 0 = The torque limits defined by parameters 20.07 Maximum torque1 and 20.08 Minimum torque1 are in force. 1 = The torque limits defined by parameters 20.09 Maximum torque2 and 20.10 Minimum torque2 are in force.	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
20.07	Maximum torque1	Defines maximum torque limit 1 for the drive (in percent of the motor nominal torque). See parameter 20.06 Torq lim sel .	
	0.0 ... 1600.0%	Maximum torque 1.	10 = 1%
20.08	Minimum torque1	Defines minimum torque limit 1 for the drive (in percent of the motor nominal torque). See parameter 20.06 Torq lim sel .	
	-1600.0 ... 0.0%	Minimum torque 1.	10 = 1%
20.09	Maximum torque2	Defines the source of maximum torque limit 2 for the drive (in percent of the motor nominal torque). See parameter 20.06 Torq lim sel .	
	AI1 scaled	02.05 AI1 scaled (see page 105).	1073742341
	AI2 scaled	02.07 AI2 scaled (see page 105).	1073742343
	FBA ref1	02.26 FBA main ref1 (see page 109).	1073742362
	FBA ref2	02.27 FBA main ref2 (see page 109).	1073742363
	D2D ref1	02.32 D2D ref1 (see page 110).	1073742368
	D2D ref2	02.33 D2D ref2 (see page 110).	1073742369
	PID out	04.05 Process PID out (see page 115).	1073742853
	Max torque1	20.07 Maximum torque1 (see page 167).	1073746951
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
20.10	Minimum torque2	Defines the source of minimum torque limit 2 for the drive (in percent of the motor nominal torque). See parameter 20.06 Torq lim sel .	
	AI1 scaled	02.05 AI1 scaled (see page 105).	1073742341
	AI2 scaled	02.07 AI2 scaled (see page 105).	1073742343
	FBA ref1	02.26 FBA main ref1 (see page 109).	1073742362
	FBA ref2	02.27 FBA main ref2 (see page 109).	1073742363
	D2D ref1	02.32 D2D ref1 (see page 110).	1073742368
	D2D ref2	02.33 D2D ref2 (see page 110).	1073742369
	PID out	04.05 Process PID out (see page 115).	1073742853
	Neg max torq	-20.09 Maximum torque2 (see page 167).	1073746949
	Min torque1	20.08 Minimum torque1 (see page 167).	1073746952
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
20.12	P motoring lim	Defines the maximum allowed power fed by the inverter to the motor in percent of the motor nominal power.	
	0.0 ... 1600.0%	Maximum motoring power.	10 = 1%
20.13	P generating lim	Defines the maximum allowed power fed by the motor to the inverter in percent of the motor nominal power.	

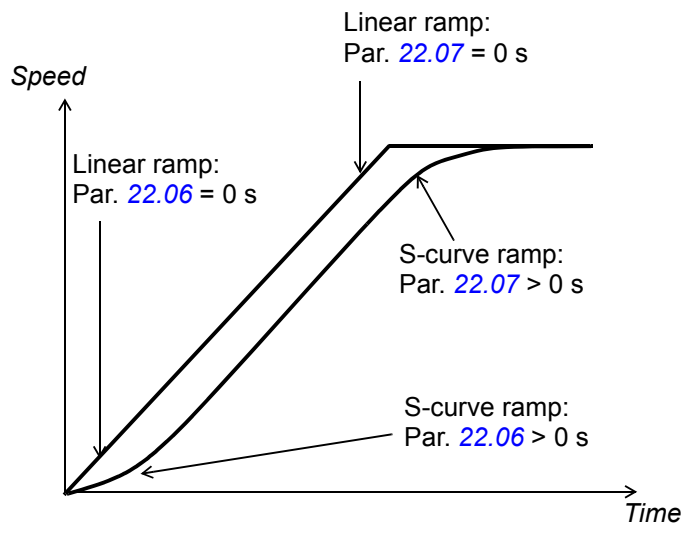
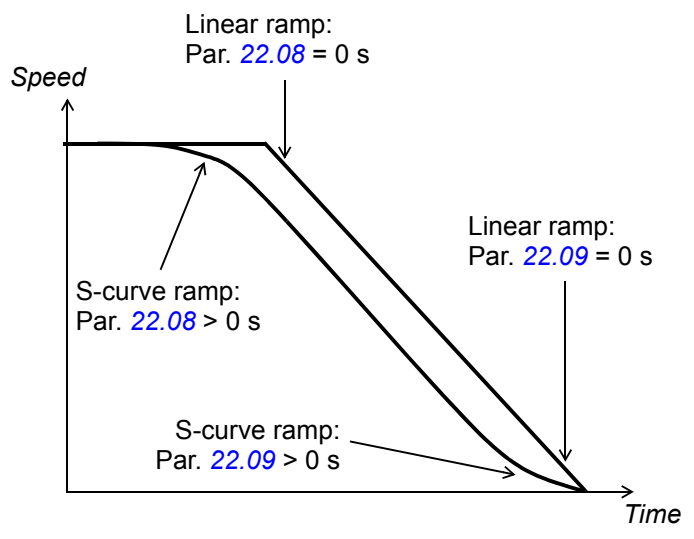
168 Parameters

No.	Name/Value	Description	FbEq
	0.0 ... 1600.0%	Maximum generating power.	10 = 1%
21 Speed ref			
		Speed reference source and scaling settings; motor potentiometer settings.	
21.01	Speed ref1 sel	Selects the source for speed reference 1. See also parameter 21.03 Speed ref1 func .	
	Zero	Zero speed reference.	0
	AI1 scaled	02.05 AI1 scaled (see page 105).	1073742341
	AI2 scaled	02.07 AI2 scaled (see page 105).	1073742343
	Freq in	02.20 Freq in (see page 105).	1073742356
	FBA ref1	02.26 FBA main ref1 (see page 109).	1073742362
	FBA ref2	02.27 FBA main ref2 (see page 109).	1073742363
	D2D ref1	02.32 D2D ref1 (see page 110).	1073742368
	D2D ref2	02.33 D2D ref2 (see page 110).	1073742369
	Panel	02.34 Panel ref (see page 110).	1073742370
	EFB ref1	02.38 EFB main ref1 (see page 114).	1073742374
	EFB ref2	02.39 EFB main ref2 (see page 114).	1073742375
	Mot pot	03.18 Speed ref pot (see page 115).	1073742610
	PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
21.02	Speed ref2 sel	Selects the source for speed reference 2.	
	Zero	Zero speed reference.	0
	AI1 scaled	02.05 AI1 scaled (see page 105).	1073742341
	AI2 scaled	02.07 AI2 scaled (see page 105).	1073742343
	Freq in	02.20 Freq in (see page 105).	1073742356
	FBA ref1	02.26 FBA main ref1 (see page 109).	1073742362
	FBA ref2	02.27 FBA main ref2 (see page 109).	1073742363
	D2D ref1	02.32 D2D ref1 (see page 110).	1073742368
	D2D ref2	02.33 D2D ref2 (see page 110).	1073742369
	Panel	02.34 Panel ref (see page 110).	1073742370
	EFB ref1	02.38 EFB main ref1 (see page 114).	1073742374
	EFB ref2	02.39 EFB main ref2 (see page 114).	1073742375
	Mot pot	03.18 Speed ref pot (see page 115).	1073742610
	PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
21.03	Speed ref1 func	Selects a mathematical function between the reference sources selected by parameters 21.01 Speed ref1 sel and 21.02 Speed ref2 sel to be used as speed reference 1.	
	Ref1	Signal selected by 21.01 Speed ref1 sel is used as speed reference 1 as such.	0
	Add	The sum of the reference sources is used as speed reference 1.	1

No.	Name/Value	Description	FbEq
	Sub	The subtraction ([21.01 Speed ref1 sel] - [21.02 Speed ref2 sel]) of the reference sources is used as speed reference 1.	2
	Mul	The multiplication of the reference sources is used as speed reference 1.	3
	Min	The smaller of the reference sources is used as speed reference 1.	4
	Max	The greater of the reference sources is used as speed reference 1.	5
21.04	Speed ref1/2 sel	Configures the selection between speed references 1 and 2. (The sources for the references are defined by parameters 21.01 Speed ref1 sel and 21.02 Speed ref2 sel respectively.) 0 = Speed reference 1 1 = Speed reference 2	
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
21.05	Speed share	Defines the scaling factor for speed reference 1/2 (speed reference 1 or 2 is multiplied by the defined value). Speed reference 1 or 2 is selected by parameter 21.04 Speed ref1/2 sel .	
	-8.000 ... 8.000	Speed reference scaling factor.	1000 = 1
21.07	Speed ref jog1	Defines the speed reference for jogging function 1. For more information on jogging, see page 85 .	
	-30000 ... 30000 rpm	Speed reference for jogging function 1.	1 = 1 rpm
21.08	Speed ref jog2	Defines the speed reference for jogging function 2. For more information on jogging, see page 85 .	
	-30000 ... 30000 rpm	Speed reference for jogging function 2.	1 = 1 rpm

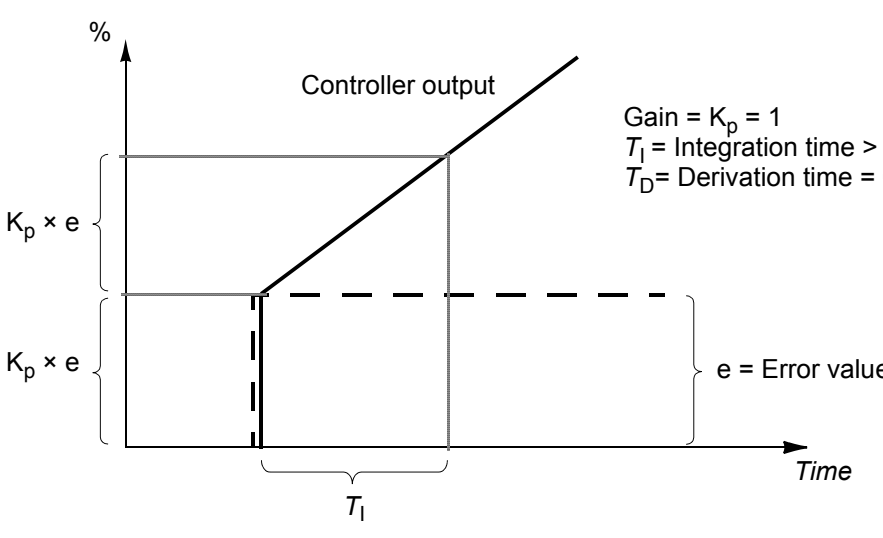
No.	Name/Value	Description	FbEq
21.09	SpeedRef min abs	Defines the absolute minimum limit for the speed reference.	
<p><i>Limited speed reference</i></p> <p>The graph illustrates the 'Limited speed reference' function. The vertical axis represents speed reference, and the horizontal axis represents speed reference. The graph shows a trapezoidal shape with a minimum speed (20.02) and a maximum speed (20.01). The absolute minimum limit for speed reference (21.09) is shown as a horizontal line. The negative of the absolute minimum limit is also shown as a horizontal line.</p>			
	0 ... 30000 rpm	Absolute minimum limit for speed reference.	1 = 1 rpm
21.10	Mot pot func	Selects whether the value of the motor potentiometer is retained upon drive power-off.	
	Reset	Drive power-off resets the value of the motor potentiometer.	0
	Store	The value of the motor potentiometer is retained over drive power-off.	1
21.11	Mot pot up	Selects the source of motor potentiometer up signal.	
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
21.12	Mot pot down	Selects the source of motor potentiometer down signal.	
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
22 Speed ref ramp		Speed reference ramp settings.	
22.01	Acc/Dec sel	Selects the source that switches between the two sets of acceleration/deceleration times defined by parameters 22.02...22.05 . 0 = Acceleration time 1 and deceleration time 1 are in force 1 = Acceleration time 2 and deceleration time 2 are in force.	
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337

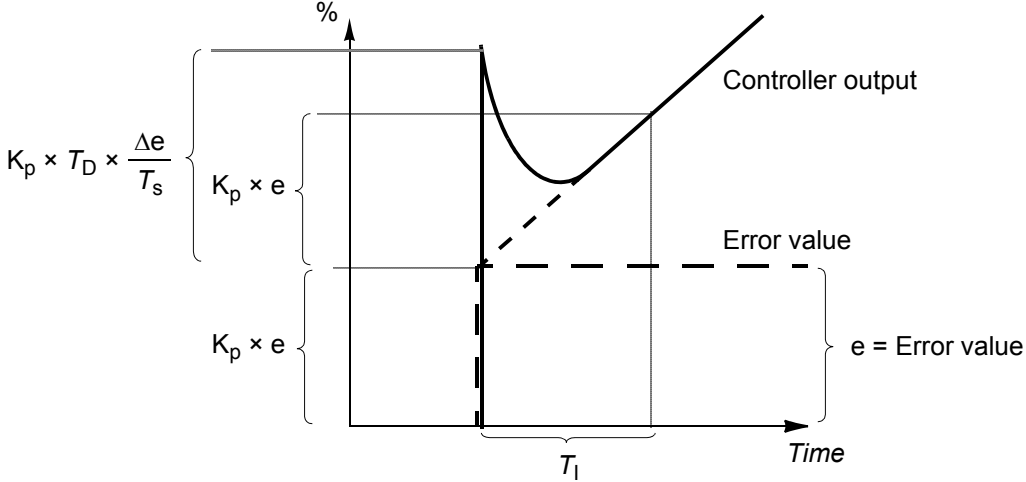
No.	Name/Value	Description	FbEq
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	Const Pointer	Bit pointer setting (see Terms and abbreviations on page 102).	-
22.02	Acc time1	<p>Defines acceleration time 1 as the time required for the speed to change from zero to the speed value defined by parameter 19.01 Speed scaling.</p> <p>If the speed reference increases faster than the set acceleration rate, the motor speed will follow the acceleration rate.</p> <p>If the speed reference increases slower than the set acceleration rate, the motor speed will follow the reference signal.</p> <p>If the acceleration time is set too short, the drive will automatically prolong the acceleration in order not to exceed the drive torque limits.</p>	
	0.000 ... 1800.000 s	Acceleration time 1.	1000 = 1 s
22.03	Dec time1	<p>Defines deceleration time 1 as the time required for the speed to change from the speed value defined by parameter 19.01 Speed scaling to zero.</p> <p>If the speed reference decreases slower than the set deceleration rate, the motor speed will follow the reference signal.</p> <p>If the reference changes faster than the set deceleration rate, the motor speed will follow the deceleration rate.</p> <p>If the deceleration time is set too short, the drive will automatically prolong the deceleration in order not to exceed drive torque limits. If there is any doubt about the deceleration time being too short, ensure that the DC overvoltage control is on (parameter 47.01 Overvolt ctrl).</p> <p>Note: If a short deceleration time is needed for a high inertia application, the drive should be equipped with an electric braking option e.g. with a brake chopper (built-in) and a brake resistor.</p>	
	0.000 ... 1800.000 s	Deceleration time 1.	1000 = 1 s
22.04	Acc time2	Defines acceleration time 2. See parameter 22.02 Acc time1 .	
	0.000 ... 1800.000 s	Acceleration time 2.	1000 = 1 s
22.05	Dec time2	Defines deceleration time 2. See parameter 22.03 Dec time1 .	
	0.000 ... 1800.000 s	Deceleration time 2.	1000 = 1 s

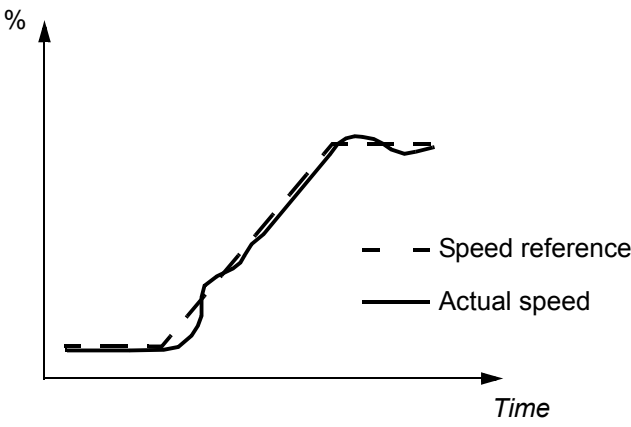
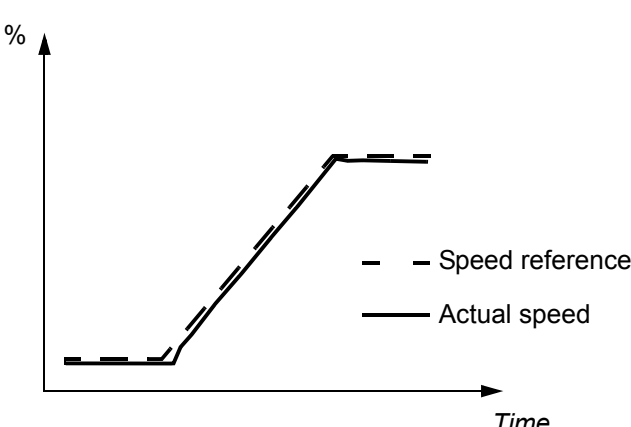
No.	Name/Value	Description	FbEq
22.06	Shape time acc1	<p>Defines the shape of the acceleration ramp at the beginning of the acceleration.</p> <p>0.000 s: Linear ramp. Suitable for steady acceleration or deceleration and for slow ramps.</p> <p>0.001...1000.000 s: S-curve ramp. S-curve ramps are ideal for lifting applications. The S-curve consists of symmetrical curves at both ends of the ramp and a linear part in between.</p> <p>Acceleration:</p>  <p>Deceleration:</p> 	
	0.000 ... 1800.000 s	Ramp shape at start of acceleration.	1000 = 1 s
22.07	Shape time acc2	Defines the shape of the acceleration ramp at the end of the acceleration. See parameter 22.06 Shape time acc1 .	
	0.000 ... 1800.000 s	Ramp shape at end of acceleration.	1000 = 1 s
22.08	Shape time dec1	Defines the shape of the deceleration ramp at the beginning of the deceleration. See parameter 22.06 Shape time acc1 .	
	0.000 ... 1800.000 s	Ramp shape at start of deceleration.	1000 = 1 s

No.	Name/Value	Description	FbEq
22.09	Shape time dec2	Defines the shape of the deceleration ramp at the end of the deceleration. See parameter 22.06 Shape time acc1 .	
	0.000 ... 1800.000 s	Ramp shape at end of deceleration.	1000 = 1 s
22.10	Acc time jogging	Defines the acceleration time for the jogging function i.e. the time required for the speed to change from zero to the speed value defined by parameter 19.01 Speed scaling .	
	0.000 ... 1800.000 s	Acceleration time for jogging.	1000 = 1 s
22.11	Dec time jogging	Defines the deceleration time for the jogging function i.e. the time required for the speed to change from the speed value defined by parameter 19.01 Speed scaling to zero.	
	0.000 ... 1800.000 s	Deceleration time for jogging.	1000 = 1 s
22.12	Em stop time	Defines the time inside which the drive is stopped if an emergency stop OFF3 is activated (i.e. the time required for the speed to change from the speed value defined by parameter 19.01 Speed scaling to zero). Emergency stop activation source is selected by parameter 10.13 Em stop off3 . Emergency stop can also be activated through fieldbus (02.22 FBA main cw or 02.36 EFB main cw). Note: Emergency stop OFF1 uses the active ramp time.	
	0.000 ... 1800.000 s	Emergency stop OFF3 deceleration time.	1000 = 1 s

23 Speed ctrl		Speed controller settings. For an autotune function, see parameter 23.20 PI tune mode .	
23.01	Proport gain	Defines the proportional gain (K_p) of the speed controller. Too large a gain may cause speed oscillation. The figure below shows the speed controller output after an error step when the error remains constant.	
<p>Gain = $K_p = 1$ T_I = Integration time = 0 T_D = Derivation time = 0</p> <p>Controller output = $K_p \times e$</p> <p>e = Error value</p>			
	0.00 ... 200.00	If gain is set to 1, a 10% change in error value (reference - actual value) causes the speed controller output to change by 10%. Note: This parameter is automatically set by the speed controller autotune function. See parameter 23.20 PI tune mode .	
		Proportional gain for speed controller.	100 = 1

No.	Name/Value	Description	FbEq
23.02	Integration time	<p>Defines the integration time of the speed controller. The integration time defines the rate at which the controller output changes when the error value is constant and the proportional gain of the speed controller is 1. The shorter the integration time, the faster the continuous error value is corrected. Too short an integration time makes the control unstable.</p> <p>If parameter value is set to zero, the I-part of the controller is disabled.</p> <p>Anti-windup stops the integrator if the controller output is limited. See 06.05 Limit word1.</p> <p>The figure below shows the speed controller output after an error step when the error remains constant.</p>  <p>Gain = $K_p = 1$ $T_I =$ Integration time > 0 $T_D =$ Derivation time $= 0$</p> <p>$K_p \times e$</p> <p>$K_p \times e$</p> <p>$e =$ Error value</p> <p>Time</p> <p>T_I</p> <p>Controller output</p>	
	0.00 ... 600.00 s	<p>Note: This parameter is automatically set by the speed controller autotune function. See parameter 23.20 PI tune mode.</p>	100 = 1 s
		Integration time for speed controller.	


No.	Name/Value	Description	FbEq
23.03	Derivation time	<p>Defines the derivation time of the speed controller. Derivative action boosts the controller output if the error value changes. The longer the derivation time, the more the speed controller output is boosted during the change. If the derivation time is set to zero, the controller works as a PI controller, otherwise as a PID controller. The derivation makes the control more responsive for disturbances.</p> <p>The speed error derivative must be filtered with a low pass filter to eliminate disturbances.</p> <p>The figure below shows the speed controller output after an error step when the error remains constant.</p>	
 <p>Gain = $K_p = 1$ T_i = Integration time > 0 T_D = Derivation time > 0 T_s = Sample time period = 250 μs Δe = Error value change between two samples</p>			
		<p>Note: Changing this parameter value is recommended only if a pulse encoder is used.</p>	
	0.000 ... 10.000 s	Derivation time for speed controller.	1000 = 1 s
23.04	Deriv filt time	Defines the derivation filter time constant. See parameter 23.03 Derivation time .	
	0.0 ... 1000.0 ms	Derivation filter time constant.	10 = 1 ms

No.	Name/Value	Description	FbEq
23.05	Acc comp DerTime	<p>Defines the derivation time for acceleration/(deceleration) compensation. In order to compensate inertia during acceleration, a derivative of the reference is added to the output of the speed controller. The principle of a derivative action is described for parameter 23.03 Derivation time.</p> <p>Note: As a general rule, set this parameter to the value between 50 and 100% of the sum of the mechanical time constants of the motor and the driven machine.</p> <p>The figure below shows the speed responses when a high inertia load is accelerated along a ramp.</p> <p>No acceleration compensation:</p>  <p>Acceleration compensation:</p> 	
	0.00 ... 600.00 s	Acceleration compensation derivation time.	100 = 1 s
23.06	Acc comp Ftime	<p>Defines the derivation filter time constant for the acceleration/(deceleration) compensation. See parameters 23.03 Derivation time and 23.05 Acc comp DerTime.</p> <p>Note: This parameter is automatically set by the speed controller autotune function (when performed in User mode). See parameter 23.20 PI tune mode.</p>	
	0.0 ... 1000.0 ms	Derivation filter time constant for acceleration compensation.	10 = 1 ms
23.07	Speed err Ftime	<p>Defines the time constant of the speed error low pass filter. If the used speed reference changes rapidly, the possible interferences in the speed measurement can be filtered with the speed error filter. Reducing the ripple with filter may cause speed controller tuning problems. A long filter time constant and fast acceleration time contradict one another. A very long filter time results in unstable control.</p>	

No.	Name/Value	Description	FbEq
	0.0 ... 1000.0 ms	Speed error filtering time constant. 0 = filtering disabled.	10 = 1 ms
23.08	Speed additive	Defines a speed reference to be added after ramping. Note: For safety reasons, the additive is not applied when stop functions are active.	
	Zero	Zero speed additive.	0
	AI1 scaled	02.05 AI1 scaled (see page 105).	1073742341
	AI2 scaled	02.07 AI2 scaled (see page 105).	1073742343
	FBA ref1	02.26 FBA main ref1 (see page 109).	1073742362
	FBA ref2	02.27 FBA main ref2 (see page 109).	1073742363
	D2D ref1	02.32 D2D ref1 (see page 110).	1073742368
	D2D ref2	02.33 D2D ref2 (see page 110).	1073742369
	PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
23.09	Max torq sp ctrl	Defines the maximum speed controller output torque.	
	-1600.0 ... 1600.0%	Maximum speed controller output torque.	10 = 1%
23.10	Min torq sp ctrl	Defines the minimum speed controller output torque.	
	-1600.0 ... 1600.0%	Minimum speed controller output torque.	10 = 1%
23.11	SpeedErr winFunc	Enables or disables speed error window control. Speed error window control forms a speed supervision function for a torque-controlled drive. It supervises the speed error value (speed reference – actual speed). In the normal operating range, window control keeps the speed controller input at zero. The speed controller is evoked only if <ul style="list-style-type: none"> the speed error exceeds the upper boundary of the window (parameter 23.12 SpeedErr win hi), or the absolute value of the negative speed error exceeds the lower boundary of the window (23.13 SpeedErr win lo). When the speed error moves outside the window, the exceeding part of the error value is connected to the speed controller. The speed controller produces a reference term relative to the input and gain of the speed controller (parameter 23.01 Proport gain) which the torque selector adds to the torque reference. The result is used as the internal torque reference for the drive. Example: In a load loss condition, the internal torque reference of the drive is decreased to prevent an excessive rise of the motor speed. If window control were inactive, the motor speed would rise until a speed limit of the drive were reached.	
	Disabled	Speed error window control inactive.	0
	Absolute	Speed error window control active. The boundaries defined by parameters 23.12 SpeedErr win hi and 23.13 SpeedErr win lo are absolute.	1
	Relative	Speed error window control active. The boundaries defined by parameters 23.12 SpeedErr win hi and 23.13 SpeedErr win lo are relative to speed reference.	2
23.12	SpeedErr win hi	Defines the upper boundary of the speed error window. Depending on setting of parameter 23.11 SpeedErr winFunc , this is either an absolute value or relative to speed reference.	
	0 ... 3000 rpm	Upper boundary of speed error window.	1 = 1 rpm

No.	Name/Value	Description	FbEq
23.13	SpeedErr win lo	Defines the lower boundary of the speed error window. Depending on setting of parameter 23.11 SpeedErr winFunc , this is either an absolute value or relative to speed reference.	
	0 ... 3000 rpm	Lower boundary of speed error window.	1 = 1 rpm
23.14	Drooping rate	<p>Defines the drooping rate (in percent of the motor nominal speed). The drooping slightly decreases the drive speed as the drive load increases. The actual speed decrease at a certain operating point depends on the drooping rate setting and the drive load (= torque reference / speed controller output). At 100% speed controller output, drooping is at its nominal level, i.e. equal to the value of this parameter. The drooping effect decreases linearly to zero along with the decreasing load.</p> <p>Droop rate can be used e.g. to adjust the load sharing in a Master/Follower application run by several drives. In a Master/Follower application the motor shafts are coupled to each other.</p> <p>The correct droop rate for a process must be found out case by case in practice.</p>	
<p>Speed decrease = Speed controller output × Drooping × Max. speed</p> <p>Example: Speed controller output is 50%, droop rate is 1%, maximum speed of the drive is 1500 rpm. Speed decrease = 0.50 × 0.01 × 1500 rpm = 7.5 rpm.</p>			
<p>The graph plots 'Motor speed in % of nominal' on the y-axis against 'Speed controller output / Drive load' on the x-axis. The y-axis has a mark at 100%. The x-axis has a mark at 100%. A solid horizontal line at the 100% level is labeled 'No drooping'. A dashed horizontal line below it is labeled 'Drooping' and '23.14 Drooping rate'. A vertical dashed line connects the 100% mark on the x-axis to the 'Drooping' line.</p>			
	0.00 ... 100.00%	Droop rate.	100 = 1%

No.	Name/Value	Description	FbEq
23.15	PI adapt max sp	<p>Maximum actual speed for speed controller adaptation. Speed controller gain and integration time can be adapted according to actual speed. This is done by multiplying the gain (23.01 Proport gain) and integration time (23.02 Integration time) by coefficients at certain speeds. The coefficients are defined individually for both gain and integration time.</p> <p>When the actual speed is below or equal to 23.16 PI adapt min sp, 23.01 Proport gain and 23.02 Integration time are multiplied by 23.17 Pcoef at min sp and 23.18 lcoef at min sp respectively.</p> <p>When the actual speed is equal to or exceeds 23.15 PI adapt max sp, no adaptation takes place; in other words, 23.01 Proport gain and 23.02 Integration time are used as such.</p> <p>Between 23.16 PI adapt min sp and 23.15 PI adapt max sp, the coefficients are calculated linearly on the basis of the breakpoints.</p>	
<p>Coefficient for K_p or T_I</p> <p>K_p = Proportional gain T_I = Integration time</p>			
	0 ... 30000 rpm	Maximum actual speed for speed controller adaptation.	1 = 1 rpm
23.16	PI adapt min sp	Minimum actual speed for speed controller adaptation. See parameter 23.15 PI adapt max sp .	
	0 ... 30000 rpm	Minimum actual speed for speed controller adaptation.	1 = 1 rpm
23.17	Pcoef at min sp	Proportional gain coefficient at minimum actual speed. See parameter 23.15 PI adapt max sp .	
	0.000 ... 10.000	Proportional gain coefficient at minimum actual speed.	1000 = 1
23.18	lcoef at min sp	Integration time coefficient at minimum actual speed. See parameter 23.15 PI adapt max sp .	
	0.000 ... 10.000	Integration time coefficient at minimum actual speed.	1000 = 1

No.	Name/Value	Description	FbEq
23.20	PI tune mode	<p>Activates the speed controller autotune function. The autotune will automatically set parameters 23.01 Proport gain and 23.02 Integration time, as well as 01.31 Mech time const. If the <i>User</i> autotune mode is chosen, also 23.07 Speed err Ftime is automatically set.</p> <p>The status of the autotune routine is shown by parameter 06.03 Speed ctrl stat.</p> <p> WARNING! The motor will reach the torque and current limits during the autotune routine. ENSURE THAT IT IS SAFE TO RUN THE MOTOR BEFORE PERFORMING THE AUTOTUNE ROUTINE!</p> <p>Notes:</p> <ul style="list-style-type: none"> • Before using the autotune function, the following parameters should be set: <ul style="list-style-type: none"> • All parameters adjusted during the start-up as described in the <i>ACS850 (Standard Control Program) Quick Start-up Guide</i> • 19.01 Speed scaling • 19.03 MotorSpeed filt • 19.06 Zero speed limit • Speed reference ramp settings in group 22 Speed ref ramp • 23.07 Speed err Ftime. <ul style="list-style-type: none"> • The drive must be in local control mode and stopped before an autotune is requested. • After requesting an autotune with this parameter, start the drive within 20 seconds. • Wait until the autotune routine is completed (this parameter has reverted to the value <i>Done</i>). The routine can be aborted by stopping the drive. • Check the values of the parameters set by the autotune function. <p>See also section Speed controller tuning on page 61.</p>	
	Done	No tuning has been requested (normal operation). The parameter also reverts to this value after an autotune is completed.	0
	Smooth	Request speed controller autotune with preset settings for smooth operation.	1
	Middle	Request speed controller autotune with preset settings for medium-tight operation.	2
	Tight	Request speed controller autotune with preset settings for tight operation.	3
	User	Request speed controller autotune with the settings defined by parameters 23.21 Tune bandwidth and 23.22 Tune damping .	4
23.21	Tune bandwidth	Speed controller bandwidth for autotune procedure, <i>User</i> mode (see parameter 23.20 PI tune mode). A larger bandwidth results in more restricted speed controller settings.	
	00.00 ... 2000.00 Hz	Tune bandwidth for <i>User</i> autotune mode.	100 = 1 Hz

No.	Name/Value	Description	FbEq
23.22	Tune damping	Speed controller damping for autotune procedure, <i>User</i> mode (see parameter 23.20 PI tune mode). Higher damping results in safer and smoother operation.	
	0.0 ... 200.0	Speed controller damping for <i>User</i> autotune mode.	10 = 1
24 Torque ref		Torque reference selection, limitation and modification settings.	
24.01	Torq ref1 sel	Selects the source for torque reference 1.	
	Zero	No torque reference selected.	0
	AI1 scaled	02.05 AI1 scaled (see page 105).	1073742341
	AI2 scaled	02.07 AI2 scaled (see page 105).	1073742343
	FBA ref1	02.26 FBA main ref1 (see page 109).	1073742362
	FBA ref2	02.27 FBA main ref2 (see page 109).	1073742363
	D2D ref1	02.32 D2D ref1 (see page 110).	1073742368
	D2D ref2	02.33 D2D ref2 (see page 110).	1073742369
	EFB ref1	02.38 EFB main ref1 (see page 114).	1073742374
	EFB ref2	02.39 EFB main ref2 (see page 114).	1073742375
	PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
24.02	Torq ref add sel	Selects the source for the torque reference addition. Because the reference is added after the torque reference selection, this parameter can be used in speed and torque control modes. Note: For safety reasons, this reference addition is not applied when stop functions are active.	
	Zero	No torque reference addition selected.	0
	AI1 scaled	02.05 AI1 scaled (see page 105).	1073742341
	AI2 scaled	02.07 AI2 scaled (see page 105).	1073742343
	FBA ref1	02.26 FBA main ref1 (see page 109).	1073742362
	FBA ref2	02.27 FBA main ref2 (see page 109).	1073742363
	D2D ref1	02.32 D2D ref1 (see page 110).	1073742368
	D2D ref2	02.33 D2D ref2 (see page 110).	1073742369
	EFB ref1	02.38 EFB main ref1 (see page 114).	1073742374
	EFB ref2	02.39 EFB main ref2 (see page 114).	1073742375
	PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
24.03	Maximum torq ref	Defines the maximum torque reference.	
	0.0 ... 1000.0%	Maximum torque reference.	10 = 1%
24.04	Minimum torq ref	Defines the minimum torque reference.	
	-1000.0 ... 0.0%	Minimum torque reference.	10 = 1%
24.05	Load share	Scales the torque reference to a required level (torque reference is multiplied by the selected value).	
	-8.000 ... 8.000	Torque reference scaling.	1000 = 1

No.	Name/Value	Description	FbEq
24.06	Torq ramp up	Defines the torque reference ramp-up time, i.e. the time for the reference to increase from zero to the nominal motor torque.	
	0.000 ... 60.000 s	Torque reference ramp-up time.	1000 = 1 s
24.07	Torq ramp down	Defines the torque reference ramp-down time, i.e. the time for the reference to decrease from the nominal motor torque to zero.	
	0.000 ... 60.000 s	Torque reference ramp-down time.	1000 = 1 s

25 Critical speed	Sets up critical speeds, or ranges of speeds, that are avoided due to, for example, mechanical resonance problems.	
--------------------------	--	--

25.01 Crit speed sel	<p>Enables/disables the critical speeds function.</p> <p>Example: A fan has vibrations in the range of 540 to 690 rpm and 1380 to 1560 rpm. To make the drive to jump over the vibration speed ranges:</p> <ul style="list-style-type: none"> • activate the critical speeds function, • set the critical speed ranges as in the figure below. <div style="text-align: center;"> </div> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 20px; text-align: center;">1</td> <td>Par. 25.02 = 540 rpm</td> </tr> <tr> <td style="text-align: center;">2</td> <td>Par. 25.03 = 690 rpm</td> </tr> <tr> <td style="text-align: center;">3</td> <td>Par. 25.04 = 1380 rpm</td> </tr> <tr> <td style="text-align: center;">4</td> <td>Par. 25.05 = 1590 rpm</td> </tr> </table>	1	Par. 25.02 = 540 rpm	2	Par. 25.03 = 690 rpm	3	Par. 25.04 = 1380 rpm	4	Par. 25.05 = 1590 rpm	
1	Par. 25.02 = 540 rpm									
2	Par. 25.03 = 690 rpm									
3	Par. 25.04 = 1380 rpm									
4	Par. 25.05 = 1590 rpm									

Disable	Critical speeds are disabled.	0
Enable	Critical speeds are enabled.	1
25.02 Crit speed1 lo	Defines the low limit for critical speed range 1. Note: This value must be less than or equal to the value of 25.03 Crit speed1 hi .	
-30000 ... 30000 rpm	Low limit for critical speed 1.	1 = 1 rpm
25.03 Crit speed1 hi	Defines the high limit for critical speed range 1. Note: This value must be greater than or equal to the value of 25.02 Crit speed1 lo .	
-30000 ... 30000 rpm	High limit for critical speed 1.	1 = 1 rpm

No.	Name/Value	Description	FbEq
25.04	Crit speed2 lo	Defines the low limit for critical speed range 2. Note: This value must be less than or equal to the value of 25.05 Crit speed2 hi .	
	-30000 ... 30000 rpm	Low limit for critical speed 2.	1 = 1 rpm
25.05	Crit speed2 hi	Defines the high limit for critical speed range 2. Note: This value must be greater than or equal to the value of 25.04 Crit speed2 lo .	
	-30000 ... 30000 rpm	High limit for critical speed 2.	1 = 1 rpm
25.06	Crit speed3 lo	Defines the low limit for critical speed range 3. Note: This value must be less than or equal to the value of 25.07 Crit speed3 hi .	
	-30000 ... 30000 rpm	Low limit for critical speed 3.	1 = 1 rpm
25.07	Crit speed3 hi	Defines the high limit for critical speed range 3. Note: This value must be greater than or equal to the value of 25.06 Crit speed3 lo .	
	-30000 ... 30000 rpm	High limit for critical speed 3.	1 = 1 rpm

26 Constant speeds

Constant speed selection and values. An active constant speed overrides the drive speed reference. See section [Constant speeds](#) on page 61.

26.01 Const speed func

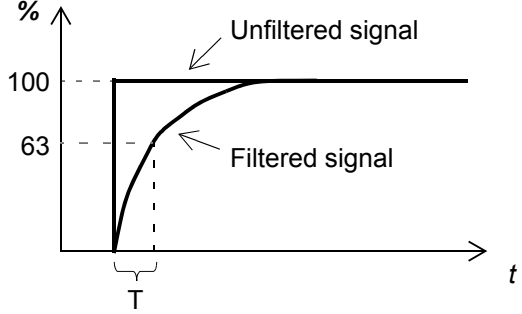
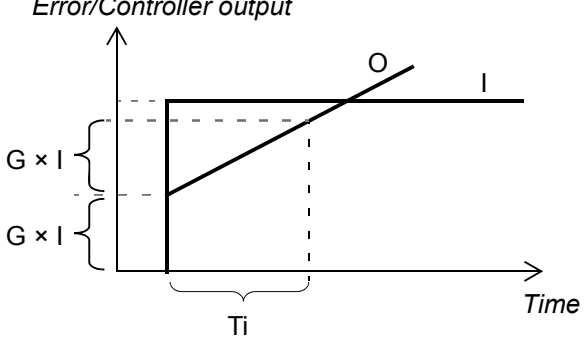
Determines how constant speeds are selected, and whether the rotation direction signal is considered or not when applying a constant speed.

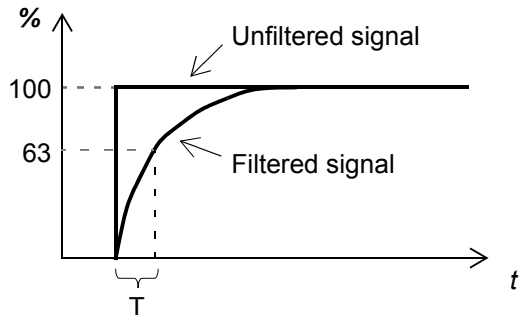
Bit	Name	Information
0	Const speed mode	1 = Packed: 7 constant speeds are selectable using the three sources defined by parameters 26.02 , 26.03 and 26.04 . 0 = Separate: Constant speeds 1, 2 and 3 are separately activated by the sources defined by parameters 26.02 , 26.03 and 26.04 respectively. In case of conflict, the constant speed with the smaller number takes priority.
1	Dir ena	1 = Start dir: To determine running direction for a constant speed, the sign of the constant speed setting (parameters 26.06...26.12) is multiplied by the direction signal (forward: +1, reverse: -1). For example, if the direction signal is reverse and the active constant speed is negative, the drive will run in the forward direction. 0 = Accord Par: The running direction for the constant speed is determined by the sign of the constant speed setting (parameters 26.06...26.12).

No.	Name/Value	Description	FbEq																																				
26.02	Const speed sel1	When bit 0 of parameter 26.01 Const speed func is 0 (Separate), selects a source that activates constant speed 1. When bit 0 of parameter 26.01 Const speed func is 1 (Packed), this parameter and parameters 26.03 Const speed sel2 and 26.04 Const speed sel3 select three sources whose states activate constant speeds as follows:																																					
		<table border="1"> <thead> <tr> <th>Source defined by par. 26.02</th> <th>Source defined by par. 26.03</th> <th>Source defined by par. 26.04</th> <th>Constant speed active</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>None</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>Constant speed 1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>Constant speed 2</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>Constant speed 3</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>Constant speed 4</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>Constant speed 5</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>Constant speed 6</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>Constant speed 7</td> </tr> </tbody> </table>	Source defined by par. 26.02	Source defined by par. 26.03	Source defined by par. 26.04	Constant speed active	0	0	0	None	1	0	0	Constant speed 1	0	1	0	Constant speed 2	1	1	0	Constant speed 3	0	0	1	Constant speed 4	1	0	1	Constant speed 5	0	1	1	Constant speed 6	1	1	1	Constant speed 7	
Source defined by par. 26.02	Source defined by par. 26.03	Source defined by par. 26.04	Constant speed active																																				
0	0	0	None																																				
1	0	0	Constant speed 1																																				
0	1	0	Constant speed 2																																				
1	1	0	Constant speed 3																																				
0	0	1	Constant speed 4																																				
1	0	1	Constant speed 5																																				
0	1	1	Constant speed 6																																				
1	1	1	Constant speed 7																																				
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337																																				
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873																																				
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409																																				
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945																																				
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481																																				
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017																																				
	Const Pointer	Bit pointer setting (see Terms and abbreviations on page 102).	-																																				
26.03	Const speed sel2	When bit 0 of parameter 26.01 Const speed func is 0 (Separate), selects a source that activates constant speed 2. When bit 0 of parameter 26.01 Const speed func is 1 (Packed), this parameter and parameters 26.02 Const speed sel1 and 26.04 Const speed sel3 select three sources that are used to activate constant speeds. See table at parameter 26.02 Const speed sel1 .																																					
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337																																				
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873																																				
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409																																				
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945																																				
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481																																				
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017																																				
	Const Pointer	Bit pointer setting (see Terms and abbreviations on page 102).	-																																				
26.04	Const speed sel3	When bit 0 of parameter 26.01 Const speed func is 0 (Separate), selects a source that activates constant speed 3. When bit 0 of parameter 26.01 Const speed func is 1 (Packed), this parameter and parameters 26.02 Const speed sel1 and 26.03 Const speed sel2 select three sources that are used to activate constant speeds. See table at parameter 26.02 Const speed sel1 .																																					
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337																																				

No.	Name/Value	Description	FbEq
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	Const Pointer	Bit pointer setting (see Terms and abbreviations on page 102).	-
26.06	Const speed1	Defines constant speed 1.	
	-30000 ... 30000 rpm	Constant speed 1.	1 = 1 rpm
26.07	Const speed2	Defines constant speed 2.	
	-30000 ... 30000 rpm	Constant speed 2.	1 = 1 rpm
26.08	Const speed3	Defines constant speed 3.	
	-30000 ... 30000 rpm	Constant speed 3.	1 = 1 rpm
26.09	Const speed4	Defines constant speed 4.	
	-30000 ... 30000 rpm	Constant speed 4.	1 = 1 rpm
26.10	Const speed5	Defines constant speed 5.	
	-30000 ... 30000 rpm	Constant speed 5.	1 = 1 rpm
26.11	Const speed6	Defines constant speed 6.	
	-30000 ... 30000 rpm	Constant speed 6.	1 = 1 rpm
26.12	Const speed7	Defines constant speed 7.	
	-30000 ... 30000 rpm	Constant speed 7.	1 = 1 rpm
27 Process PID		Configuration of process PID control. See also section Process PID control on page 70 .	
27.01	PID setpoint sel	Selects the source of setpoint (reference) for the PID controller.	
	Zero	Zero reference.	0
	AI1 scaled	02.05 AI1 scaled (see page 105).	1073742341
	AI2 scaled	02.07 AI2 scaled (see page 105).	1073742343
	FBA ref1	02.26 FBA main ref1 (see page 109).	1073742362
	FBA ref2	02.27 FBA main ref2 (see page 109).	1073742363
	D2D ref1	02.32 D2D ref1 (see page 110).	1073742368
	D2D ref2	02.33 D2D ref2 (see page 110).	1073742369
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
27.02	PID fbk func	Defines how the final process feedback is calculated from the two sources selected by parameters 27.03 PID fbk1 src and 27.04 PID fbk2 src .	
	Act1	Process feedback 1 used.	0



No.	Name/Value	Description	FbEq
	Add	Sum of feedback 1 and feedback 2.	1
	Sub	Feedback 2 subtracted from feedback 1.	2
	Mul	Feedback 1 multiplied by feedback 2.	3
	div	Feedback 1 divided by feedback 2.	4
	Max	Greater of the two feedback sources used.	5
	Min	Smaller of the two feedbacks sources used.	6
	Sqrt sub	Square root of (feedback 1 – feedback 2).	7
	Sqrt add	Square root of feedback 1 + square root of feedback 2.	8
27.03	PID fbk1 src	Selects the source of process feedback 1.	
	Zero	Zero feedback.	0
	AI1 scaled	02.05 AI1 scaled (see page 105).	1073742341
	AI2 scaled	02.07 AI2 scaled (see page 105).	1073742343
	FBA ref1	02.26 FBA main ref1 (see page 109).	1073742362
	FBA ref2	02.27 FBA main ref2 (see page 109).	1073742363
	D2D ref1	02.32 D2D ref1 (see page 110).	1073742368
	D2D ref2	02.33 D2D ref2 (see page 110).	1073742369
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
27.04	PID fbk2 src	Selects the source of process feedback 2.	
	Zero	Zero feedback.	0
	AI1 scaled	02.05 AI1 scaled (see page 105).	1073742341
	AI2 scaled	02.07 AI2 scaled (see page 105).	1073742343
	FBA ref1	02.26 FBA main ref1 (see page 109).	1073742362
	FBA ref2	02.27 FBA main ref2 (see page 109).	1073742363
	D2D ref1	02.32 D2D ref1 (see page 110).	1073742368
	D2D ref2	02.33 D2D ref2 (see page 110).	1073742369
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
27.05	PID fbk1 max	Maximum value for process feedback 1.	
	-32768.00 ... 32768.00	Maximum value for process feedback 1.	100 = 1
27.06	PID fbk1 min	Minimum value for process feedback 1.	
	-32768.00 ... 32768.00	Minimum value for process feedback 1.	100 = 1
27.07	PID fbk2 max	Maximum value for process feedback 2.	
	-32768.00 ... 32768.00	Maximum value for process feedback 2.	100 = 1
27.08	PID fbk2 min	Minimum value for process feedback 2.	
	-32768.00 ... 32768.00	Minimum value for process feedback 2.	100 = 1
27.09	PID fbk gain	Multiplier for scaling the final feedback value for process PID controller.	
	-32.768 ... 32.767	PID feedback gain.	1000 = 1

No.	Name/Value	Description	FbEq
27.10	PID fbk ftime	Defines the time constant for the filter through which the process feedback is connected to the PID controller.	
	0.000 ... 30.000 s	Filter time constant.  $O = I \times (1 - e^{-t/T})$ <p> <i>I</i> = filter input (step) <i>O</i> = filter output <i>t</i> = time <i>T</i> = filter time constant </p>	1000 = 1 s
27.12	PID gain	Defines the gain for the process PID controller. See parameter 27.13 PID integ time .	
	0.00 ... 100.00	Gain for PID controller.	100 = 1
27.13	PID integ time	Defines the integration time for the process PID controller.	
		 <p> <i>I</i> = controller input (error) <i>O</i> = controller output <i>G</i> = gain <i>Ti</i> = integration time </p> <p>Note: Setting this value to 0 disables the “I” part, turning the PID controller into a PD controller.</p>	
	0.00 ... 320.00 s	Integration time.	100 = 1 s
27.14	PID deriv time	Defines the derivation time of the process PID controller. The derivative component at the controller output is calculated on basis of two consecutive error values (E_{K-1} and E_K) according to the following formula: $\text{PID DERIV TIME} \times (E_K - E_{K-1})/T_S$, in which $T_S = 12 \text{ ms}$ sample time $E = \text{Error} = \text{Process reference} - \text{process feedback}$.	
	0.00 ... 10.00 s	Derivation time.	100 = 1 s


No.	Name/Value	Description	FbEq
27.15	PID deriv filter	<p>Defines the time constant of the 1-pole filter used to smooth the derivative component of the process PID controller.</p>  <p>$O = I \times (1 - e^{-t/T})$</p> <p>I = filter input (step) O = filter output t = time T = filter time constant</p>	
	0.00 ... 10.00 s	Filter time constant.	100 = 1 s
27.16	PID error inv	PID error inversion. When the source selected by this parameter is on, the error (process setpoint – process feedback) at the PID controller input is inverted.	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
27.17	PID mode	Activates the PID controller output trimming function. Using the trim it is possible to apply a correction factor to the drive reference.	
	Direct	Proportional trimming not used.	0
	Prop speed	PID controller output is trimmed in proportion to speed.	1
	Prop torque	PID controller output is trimmed in proportion to torque.	2
27.18	PID maximum	Defines the maximum limit for the PID controller output. Using the minimum and maximum limits, it is possible to restrict the operation range.	
	-32768.0 ... 32768.0	Maximum limit for PID controller output.	10 = 1
27.19	PID minimum	Defines the minimum limit for the PID controller output. See parameter 27.18 PID maximum .	
	-32768.0 ... 32768.0	Minimum limit for PID controller output.	10 = 1
27.22	Sleep mode	Activates the sleep function.	
	No	Sleep function inactive.	0
	Internal	The sleep function is activated and deactivated automatically as defined by parameters 27.23 Sleep level and 27.24 Sleep delay . The sleep and wake-up delays (27.24 Sleep delay and 27.26 Wake up delay) are effective.	1
	External	The sleep function is activated by the source selected by parameter 27.27 Sleep ena . The sleep and wake-up delays (27.24 Sleep delay and 27.26 Wake up delay) are effective.	2

No.	Name/Value	Description	FbEq
27.23	Sleep level	Defines the start limit for the sleep function. If the motor speed is below this value longer than the sleep delay (27.24 Sleep delay), the drive shifts to sleep mode.	
	-32768.0 ... 32768.0	Sleep start level.	10 = 1
27.24	Sleep delay	Defines the delay for the sleep start function. See parameter 27.23 Sleep level . When the motor speed falls below the sleep level, the counter starts. When the motor speed exceeds the sleep level, the counter resets.	
	0.0 ... 360.0 s	Sleep start delay.	10 = 1 s
27.25	Wake up level	Defines the wake-up limit for the sleep function. The drive wakes up if the process actual value is above the wake-up level longer than the wake-up delay (27.26 Wake up delay).	
	0.0 ... 32768.0	Wake-up level.	10 = 1
27.26	Wake up delay	Defines the wake-up delay for the sleep function. See parameter 27.25 Wake up level . When the process actual value falls below the wake-up level, the wake-up counter starts. When the process actual value exceeds the wake-up level, the counter resets.	
	0.0 ... 360.0 s	Wake-up delay.	10 = 1 s
27.27	Sleep ena	Defines a source that can be used to activate sleep mode when parameter 27.22 Sleep mode is set to <i>External</i> .	
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
30 Fault functions		Selects the behavior of the drive upon various fault situations.	
30.01	External fault	Selects an source for an external fault signal. 0 = External fault trip 1 = No external fault	
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019


190 Parameters

No.	Name/Value	Description	FbEq
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
30.02	Speed ref safe	Defines the safe speed reference that is used with the Spd ref Safe setting of supervision parameters 13.32 AI superv func , 30.03 Local ctrl loss or 50.02 Comm loss func upon an alarm. This speed is used when the parameter is set to Spd ref Safe .	
	-30000 ... 30000 rpm	Safe speed reference.	1 = 1 rpm
30.03	Local ctrl loss	Selects how the drive reacts to a control panel or PC tool communication break.	
	No	No action taken.	0
	Fault	Drive trips on fault LOCAL CTRL LOSS.	1
	Spd ref Safe	The drive generates alarm LOCAL CTRL LOSS and sets the speed to the speed defined by parameter 30.02 Speed ref safe .  WARNING! Make sure that it is safe to continue operation in case of a communication break.	2
	Last speed	The drive generates alarm LOCAL CTRL LOSS and freezes the speed to the level the drive was operating at. The speed is determined by the average speed over the previous 10 seconds.  WARNING! Make sure that it is safe to continue operation in case of a communication break.	3
30.04	Mot phase loss	Selects how the drive reacts when a motor phase loss is detected.	
	No	No action taken.	0
	Fault	The drive trips on fault MOTOR PHASE.	1
30.05	Earth fault	Selects how the drive reacts when an earth fault or current unbalance is detected in the motor or the motor cable.	
	No	No action taken.	0
	Warning	The drive generates alarm EARTH FAULT.	1
	Fault	The drive trips on fault EARTH FAULT.	2
30.06	Suppl phs loss	Selects how the drive reacts when a supply phase loss is detected.	
	No	No action taken.	0
	Fault	The drive trips on fault SUPPLY PHASE.	1
30.07	Sto diagnostic	Selects how the drive reacts when it detects the absence of one or both Safe torque off (STO) signals. Note: This parameter is for supervision only. The Safe torque off function can activate even when this parameter is set to No . For general information on the Safe torque off function, see the <i>Hardware manual</i> of the drive, and <i>Application guide - Safe torque off function for ACSM1, ACS850 and ACQ810 drives</i> (3AFE68929814 [English]).	
	Fault	The drive trips on SAFE TORQUE OFF when one or both of the STO signals are lost.	1

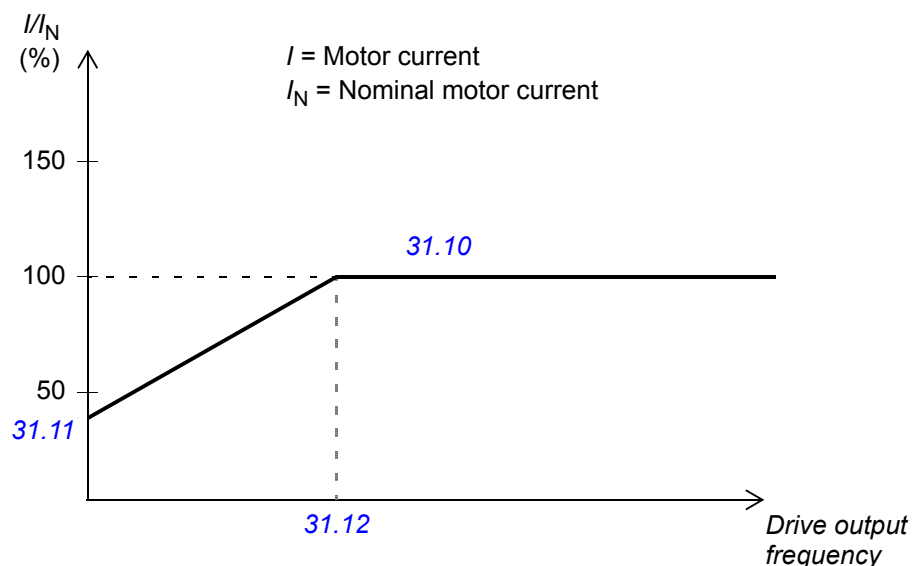
No.	Name/Value	Description	FbEq								
	Alarm	<u>Drive running:</u> The drive trips on SAFE TORQUE OFF when one or both of the STO signals is lost. <u>Drive stopped:</u> The drive generates a SAFE TORQUE OFF alarm if both STO signals are absent. If only one of the signals is lost, the drive trips on STO1 LOST or STO2 LOST.	2								
	No	<u>Drive running:</u> The drive trips on SAFE TORQUE OFF when one or both of the STO signals is lost. <u>Drive stopped:</u> No action if both STO signals are absent. If only one of the signals is lost, the drive trips on STO1 LOST or STO2 LOST.	3								
	Only Alarm	The drive generates a SAFE TORQUE OFF alarm if both STO signals are absent. If only one of the signals is lost, the drive trips on STO1 LOST or STO2 LOST.	4								
30.08	Cross connection	Selects how the drive reacts to incorrect input power and motor cable connection (i.e. input power cable is connected to drive motor connection).									
	No	No action taken.	0								
	Fault	The drive trips on fault CABLE CROSS CON.	1								
30.09	Stall function	Selects how the drive reacts to a motor stall condition. A stall condition is defined as follows: <ul style="list-style-type: none"> • The drive is at stall current limit (30.10 Stall curr lim), and • the output frequency is below the level set by parameter 30.11 Stall freq hi, and • the conditions above have been valid longer than the time set by parameter 30.12 Stall time. See section Stall protection (parameters 30.09...30.12) on page 84 .									
<table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Ena sup (Enable supervision) 0 = Disabled: Supervision disabled. 1 = Enabled: Supervision enabled.</td> </tr> <tr> <td>1</td> <td>Ena warn (Enable warning) 0 = Disabled 1 = Enabled: Drive generates an alarm upon a stall condition.</td> </tr> <tr> <td>2</td> <td>Ena fault (Enable fault) 0 = Disabled 1 = Enabled: Drive trips on a fault upon a stall condition.</td> </tr> </tbody> </table>				Bit	Function	0	Ena sup (Enable supervision) 0 = Disabled: Supervision disabled. 1 = Enabled: Supervision enabled.	1	Ena warn (Enable warning) 0 = Disabled 1 = Enabled: Drive generates an alarm upon a stall condition.	2	Ena fault (Enable fault) 0 = Disabled 1 = Enabled: Drive trips on a fault upon a stall condition.
Bit	Function										
0	Ena sup (Enable supervision) 0 = Disabled: Supervision disabled. 1 = Enabled: Supervision enabled.										
1	Ena warn (Enable warning) 0 = Disabled 1 = Enabled: Drive generates an alarm upon a stall condition.										
2	Ena fault (Enable fault) 0 = Disabled 1 = Enabled: Drive trips on a fault upon a stall condition.										
30.10	Stall curr lim	Stall current limit in percent of the nominal current of the motor. See parameter 30.09 Stall function .									
	0.0 ... 1600.0%	Stall current limit.	10 = 1%								
30.11	Stall freq hi	Stall frequency limit. See parameter 30.09 Stall function . Note: Setting the limit below 10 Hz is not recommended.									
	0.5 ... 1000.0 Hz	Stall frequency limit.	10 = 1 Hz								
30.12	Stall time	Stall time. See parameter 30.09 Stall function .									
	0 ... 3600 s	Stall time.	1 = 1 s								

No.	Name/Value	Description	FbEq
31	Motor therm prot	Motor temperature measurement and thermal protection settings.	
31.01	Mot temp1 prot	Selects how the drive reacts when motor overtemperature is detected by motor thermal protection 1.	
	No	Motor thermal protection 1 inactive.	0
	Alarm	The drive generates alarm MOTOR TEMPERATURE when the temperature exceeds the alarm level defined by parameter <i>31.03 Mot temp1 almLim</i> .	1
	Fault	The drive generates alarm MOTOR TEMPERATURE or trips on fault MOTOR OVERTEMP when the temperature exceeds the alarm/fault level defined by parameter <i>31.03 Mot temp1 almLim</i> / <i>31.04 Mot temp1 fltLim</i> (whichever is lower).	2
31.02	Mot temp1 src	Selects the means of temperature measurement for motor thermal protection 1. When overtemperature is detected the drive reacts as defined by parameter <i>31.01 Mot temp1 prot</i> . Note: If one FEN-xx module is used, parameter setting must be either KTY 1st FEN or PTC 1st FEN. The FEN-xx module can be in either Slot 1 or Slot 2.	
	Estimated	The temperature is supervised based on the motor thermal protection model, which uses the motor thermal time constant (parameter <i>31.14 Mot therm time</i>) and the motor load curve (parameters <i>31.10...31.12</i>). User tuning is typically needed only if the ambient temperature differs from the normal operating temperature specified for the motor. The motor temperature increases if it operates in the region above the motor load curve. The motor temperature decreases if it operates in the region below the motor load curve (if the motor is overheated).  WARNING! The model does not protect the motor if it does not cool properly due to dust and dirt.	0
	KTY 1st FEN	The temperature is supervised using a KTY84 sensor connected to encoder interface module FEN-xx installed in drive Slot 1/2. If two encoder interface modules are used, encoder module connected to Slot 1 is used for the temperature supervision. Note: This selection does not apply to FEN-01.	2
	KTY 2nd FEN	The temperature is supervised using a KTY84 sensor connected to encoder interface module FEN-xx installed in drive Slot 1/2. If two encoder interface modules are used, encoder module connected to Slot 2 is used for the temperature supervision. Note: This selection does not apply to FEN-01.	3
	PTC JCU	The temperature is supervised using a PTC sensor connected to DI6.	4
	PTC 1st FEN	The temperature is supervised using a PTC sensor connected to encoder interface module FEN-xx installed in drive Slot 1/2. If two encoder interface modules are used, encoder module connected to Slot 1 is used for the temperature supervision.	5

No.	Name/Value	Description	FbEq
	PTC 2nd FEN	The temperature is supervised using a PTC sensor connected to encoder interface module FEN-xx installed in drive Slot 1/2. If two encoder interface modules are used, encoder module connected to Slot 2 is used for the temperature supervision.	6
	Pt100 JCU x1	The temperature is supervised using a Pt100 sensor connected to analog input AI1 and analog output AO1 on the JCU Control Unit of the drive. The analog output feeds constant current through the sensor. The sensor resistance increases as the motor temperature rises, as does the voltage over the sensor. The temperature measurement function reads the voltage through the analog input and converts it to degrees centigrade.	7
	Pt100 JCU x2	The temperature is supervised using two Pt100 sensors connected to analog input AI1 and analog output AO1 on the JCU Control Unit of the drive. See Pt100 JCU x1 above.	8
	Pt100 JCU x3	The temperature is supervised using three Pt100 sensors connected to analog input AI1 and analog output AO1 on the JCU Control Unit of the drive. See Pt100 JCU x1 above.	9
	Pt100 Ext x1	The temperature is supervised using a Pt100 sensor connected to the first available analog input and analog output on I/O extensions installed on the drive. See Pt100 JCU x1 above.	10
	Pt100 Ext x2	The temperature is supervised using two Pt100 sensors connected to the first available analog input and analog output on I/O extensions installed on the drive. See Pt100 JCU x1 above.	11
	Pt100 Ext x3	The temperature is supervised using three Pt100 sensors connected to the first available analog input and analog output on I/O extensions installed on the drive. See Pt100 JCU x1 above.	12
31.03	Mot temp1 almLim	Defines the alarm limit for motor thermal protection 1 (when parameter 31.01 Mot temp1 prot is set to either <i>Alarm</i> or <i>Fault</i>).	
	0 ... 200 °C	Motor overtemperature alarm limit.	1 = 1 °C
31.04	Mot temp1 fItLim	Defines the fault limit for the motor thermal protection 1 (when parameter 31.01 Mot temp1 prot is set to <i>Fault</i>).	
	0 ... 200 °C	Motor overtemperature fault limit.	1 = 1 °C
31.05	Mot temp2 prot	Selects how the drive reacts when motor overtemperature is detected by motor temperature protection 2.	
	No	Motor temperature protection 2 inactive.	0
	Alarm	The drive generates alarm MOTTEMPAL2 when the temperature exceeds the alarm level defined by parameter 31.07 Mot temp2 almLim .	1
	Fault	The drive generates alarm MOTTEMPAL2 or trips on fault MOTTEMP2 when the temperature exceeds the alarm/fault level defined by parameter 31.07 Mot temp2 almLim / 31.08 Mot temp2 fItLim (whichever is lower).	2

No.	Name/Value	Description	FbEq
31.06	Mot temp2 src	<p>Selects the means of temperature measurement for motor thermal protection 2. When overtemperature is detected the drive reacts as defined by parameter 31.05 Mot temp2 prot.</p> <p>Note: If one FEN-xx module is used, parameter setting must be either KTY 1st FEN or PTC 1st FEN. The FEN-xx module can be in either Slot 1 or Slot 2.</p>	
	Estimated	<p>The temperature is supervised based on the motor thermal protection model, which uses the motor thermal time constant (parameter 31.14 Mot therm time) and the motor load curve (parameters 31.10...31.12). User tuning is typically needed only if the ambient temperature differs from the normal operating temperature specified for the motor.</p> <p>The motor temperature increases if it operates in the region above the motor load curve. The motor temperature decreases if it operates in the region below the motor load curve (if the motor is overheated).</p> <p> WARNING! The model does not protect the motor if it does not cool properly due to dust and dirt.</p>	0
	KTY 1st FEN	<p>The temperature is supervised using a KTY84 sensor connected to encoder interface module FEN-xx installed in drive Slot 1/2. If two encoder interface modules are used, encoder module connected to Slot 1 is used for the temperature supervision.</p> <p>Note: This selection does not apply to FEN-01.</p>	2
	KTY 2nd FEN	<p>The temperature is supervised using a KTY84 sensor connected to encoder interface module FEN-xx installed in drive Slot 1/2. If two encoder interface modules are used, encoder module connected to Slot 2 is used for the temperature supervision.</p> <p>Note: This selection does not apply to FEN-01.</p>	3
	PTC JCU	<p>The temperature is supervised using a PTC sensor connected to DI6.</p>	4
	PTC 1st FEN	<p>The temperature is supervised using a PTC sensor connected to encoder interface module FEN-xx installed in drive Slot 1/2. If two encoder interface modules are used, encoder module connected to Slot 1 is used for the temperature supervision.</p>	5
	PTC 2nd FEN	<p>The temperature is supervised using a PTC sensor connected to encoder interface module FEN-xx installed in drive Slot 1/2. If two encoder interface modules are used, encoder module connected to Slot 2 is used for the temperature supervision.</p>	6
	Pt100 JCU x1	<p>The temperature is supervised using one Pt100 sensor connected to analog input AI1 and analog output AO1 on the JCU Control Unit of the drive.</p> <p>The analog output feeds constant current through the sensor. The sensor resistance increases as the motor temperature rises, as does the voltage over the sensor. The temperature measurement function reads the voltage through the analog input and converts it to degrees centigrade.</p>	7
	Pt100 JCU x2	<p>The temperature is supervised using two Pt100 sensors connected to analog input AI1 and analog output AO1 on the JCU Control Unit of the drive. See Pt100 JCU x1 above.</p>	8

No.	Name/Value	Description	FbEq
	Pt100 JCU x3	The temperature is supervised using three Pt100 sensors connected to analog input AI1 and analog output AO1 on the JCU Control Unit of the drive. See Pt100 JCU x1 above.	9
	Pt100 Ext x1	The temperature is supervised using a Pt100 sensor connected to the first available analog input and analog output on I/O extensions installed on the drive. See Pt100 JCU x1 above.	10
	Pt100 Ext x2	The temperature is supervised using two Pt100 sensors connected to the first available analog input and analog output on I/O extensions installed on the drive. See Pt100 JCU x1 above.	11
	Pt100 Ext x3	The temperature is supervised using three Pt100 sensors connected to the first available analog input and analog output on I/O extensions installed on the drive. See Pt100 JCU x1 above.	12
31.07	Mot temp2 almLim	Defines the alarm limit for the motor thermal protection 2 (when parameter 31.05 Mot temp2 prot is set to either <i>Alarm</i> or <i>Fault</i>).	
	0 ... 200 °C	Motor overtemperature alarm limit.	1 = 1 °C
31.08	Mot temp2 fltLim	Defines the fault limit for the motor thermal protection 2 (when parameter 31.05 Mot temp2 prot is set to <i>Fault</i>).	
	0 ... 200 °C	Motor overtemperature fault limit.	1 = 1 °C
31.09	Mot ambient temp	Defines the ambient temperature for the thermal protection mode.	
	-60 ... 100 °C	Ambient temperature.	1 = 1 °C
31.10	Mot load curve	Defines the load curve together with parameters 31.11 Zero speed load and 31.12 Break point When the parameter is set to 100%, the maximum load is equal to the value of parameter 99.06 Mot nom current (higher loads heat up the motor). The load curve level should be adjusted if the ambient temperature differs from the nominal value. The load curve is used by the motor thermal protection model when parameter 31.02 Mot temp1 src is set to <i>Estimated</i> .	



No.	Name/Value	Description	FbEq
	50 ... 150%	Maximum load for the motor load curve.	1 = 1%
31.11	Zero speed load	Defines the motor load curve together with parameters 31.10 Mot load curve and 31.12 Break point . Defines the maximum motor load at zero speed of the load curve. A higher value can be used if the motor has an external motor fan to boost the cooling. See the motor manufacturer's recommendations. See parameter 31.10 Mot load curve .	
	50 ... 150%	Zero speed load for the motor load curve.	1 = 1%
31.12	Break point	Defines the motor load curve together with parameters 31.10 Mot load curve and 31.11 Zero speed load . Defines the break point frequency of the load curve i.e. the point at which the motor load curve begins to decrease from the value of parameter 31.10 Mot load curve towards the value of parameter 31.11 Zero speed load . See parameter 31.10 Mot load curve .	
	0.01 ... 500.00 Hz	Break point for the motor load curve.	100 = 1 Hz
31.13	Mot nom tempRise	Defines the temperature rise of the motor when the motor is loaded with nominal current. See the motor manufacturer's recommendations. The temperature rise value is used by the motor thermal protection model when parameter 31.02 Mot temp1 src is set to <i>Estimated</i> .	
	0 ... 300 °C	Temperature rise.	1 = 1 °C

No.	Name/Value	Description	FbEq														
31.14	Mot therm time	<p>Defines the thermal time constant for the motor thermal protection model (i.e. time inside which the temperature has reached 63% of the nominal temperature). See the motor manufacturer's recommendations.</p> <p>The motor thermal protection model is used when parameter 31.02 Mot temp1 src is set to <i>Estimated</i>.</p>															
	100 ... 10000 s	Motor thermal time constant.	1 = 1 s														
32 Automatic reset		Defines conditions for automatic fault resets.															
32.01	Autoreset sel	<p>Selects faults that are automatically reset. The parameter is a 16-bit word with each bit corresponding to a fault type. Whenever a bit is set to 1, the corresponding fault is automatically reset.</p> <p>The bits of the binary number correspond to the following faults:</p>															
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">Bit</th> <th style="text-align: left;">Fault</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Overcurrent</td> </tr> <tr> <td>1</td> <td>Overvoltage</td> </tr> <tr> <td>2</td> <td>Undervoltage</td> </tr> <tr> <td>3</td> <td>AI min</td> </tr> <tr> <td>4</td> <td>Reserved</td> </tr> <tr> <td>5</td> <td>External fault</td> </tr> </tbody> </table>				Bit	Fault	0	Overcurrent	1	Overvoltage	2	Undervoltage	3	AI min	4	Reserved	5	External fault
Bit	Fault																
0	Overcurrent																
1	Overvoltage																
2	Undervoltage																
3	AI min																
4	Reserved																
5	External fault																
32.02	Number of trials	Defines the number of automatic fault resets the drive performs within the time defined by parameter 32.03 Trial time .															
	0 ... 5	Number of automatic resets.	1 = 1														
32.03	Trial time	Defines the time for the automatic fault reset function. See parameter 32.02 Number of trials .															
	1.0 ... 600.0 s	Time for automatic resets.	10 = 1 s														

No.	Name/Value	Description	FbEq
32.04	Delay time	Defines the time that the drive will wait after a fault before attempting an automatic reset. See parameter 32.01 Autoreset sel.	
	0.0 ... 120.0 s	Resetting delay.	10 = 1 s
33 Supervision		Configuration of signal supervision. See also section Signal supervision on page 84.	
33.01	Superv1 func	Selects the mode of supervision 1.	
	Disabled	Supervision 1 not in use.	0
	Low	When the signal selected by parameter 33.02 Superv1 act falls below the value of parameter 33.04 Superv1 lo , bit 0 of 06.13 Superv status is activated. To clear the bit, the signal must exceed the value of parameter 33.03 Superv1 hi .	1
	High	When the signal selected by parameter 33.02 Superv1 act exceeds the value of parameter 33.03 Superv1 hi , bit 0 of 06.13 Superv status is activated. To clear the bit, the signal must fall below the value of parameter 33.04 Superv1 lo .	2
	Abs Low	When the absolute value of the signal selected by parameter 33.02 Superv1 act falls below the value of parameter 33.04 Superv1 lo , bit 0 of 06.13 Superv status is activated. To clear the bit, the absolute value of the signal must exceed the value of parameter 33.03 Superv1 hi .	3
	Abs High	When the absolute value of the signal selected by parameter 33.02 Superv1 act exceeds the value of parameter 33.03 Superv1 hi , bit 0 of 06.13 Superv status is activated. To clear the bit, the absolute value of the signal must fall below the value of parameter 33.04 Superv1 lo .	4
33.02	Superv1 act	Selects the signal to be monitored by supervision 1. See parameter 33.01 Superv1 func .	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	SpRef unramp	03.03 SpeedRef unramp (see page 114).	1073742595
	SpRef ramped	03.05 SpeedRef ramped (see page 114).	1073742597
	SpRef used	03.06 SpeedRef used (see page 114).	1073742598
	TorqRef used	03.14 Torq ref used (see page 115).	1073742606
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-

No.	Name/Value	Description	FbEq
33.03	Superv1 hi	Selects the upper limit for supervision 1. See parameter 33.01 Superv1 func.	
	-32768.00 ... 32768.00	Upper limit for supervision 1.	100 = 1
33.04	Superv1 lo	Selects the lower limit for supervision 1. See parameter 33.01 Superv1 func.	
	-32768.00 ... 32768.00	Lower limit for supervision 1.	100 = 1
33.05	Superv2 func	Selects the mode of supervision 2.	
	Disabled	Supervision 2 not in use.	0
	Low	When the signal selected by parameter 33.06 Superv2 act falls below the value of parameter 33.08 Superv2 lo , bit 1 of 06.13 Superv status is activated. To clear the bit, the signal must exceed the value of parameter 33.07 Superv2 hi .	1
	High	When the signal selected by parameter 33.06 Superv2 act exceeds the value of parameter 33.07 Superv2 hi , bit 1 of 06.13 Superv status is activated. To clear the bit, the signal must fall below the value of parameter 33.08 Superv2 lo .	2
	Abs Low	When the absolute value of the signal selected by parameter 33.06 Superv2 act falls below the value of parameter 33.08 Superv2 lo , bit 1 of 06.13 Superv status is activated. To clear the bit, the absolute value of the signal must exceed the value of parameter 33.07 Superv2 hi .	3
	Abs High	When the absolute value of the signal selected by parameter 33.06 Superv2 act exceeds the value of parameter 33.07 Superv2 hi , bit 1 of 06.13 Superv status is activated. To clear the bit, the absolute value of the signal must fall below the value of parameter 33.08 Superv2 lo .	4
33.06	Superv2 act	Selects the signal to be monitored by supervision 2. See parameter 33.05 Superv2 func.	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	SpRef unramp	03.03 SpeedRef unramp (see page 114).	1073742595
	SpRef ramped	03.05 SpeedRef ramped (see page 114).	1073742597
	SpRef used	03.06 SpeedRef used (see page 114).	1073742598
	TorqRef used	03.14 Torq ref used (see page 115).	1073742606
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-

200 Parameters

No.	Name/Value	Description	FbEq
33.07	Superv2 hi	Selects the upper limit for supervision 2. See parameter 33.05 Superv2 func.	
	-32768.00 ... 32768.00	Upper limit for supervision 2.	100 = 1
33.08	Superv2 lo	Selects the lower limit for supervision 2. See parameter 33.05 Superv2 func.	
	-32768.00 ... 32768.00	Lower limit for supervision 2.	100 = 1
33.09	Superv3 func	Selects the mode of supervision 3.	
	Disabled	Supervision 3 not in use.	0
	Low	When the signal selected by parameter 33.10 Superv3 act falls below the value of parameter 33.12 Superv3 lo , bit 2 of 06.13 Superv status is activated. To clear the bit, the signal must exceed the value of parameter 33.11 Superv3 hi .	1
	High	When the signal selected by parameter 33.10 Superv2 act exceeds the value of parameter 33.11 Superv3 hi , bit 2 of 06.13 Superv status is activated. To clear the bit, the signal must fall below the value of parameter 33.12 Superv3 lo .	2
	Abs Low	When the absolute value of the signal selected by parameter 33.10 Superv3 act falls below the value of parameter 33.12 Superv3 lo , bit 2 of 06.13 Superv status is activated. To clear the bit, the absolute value of the signal must exceed the value of parameter 33.11 Superv3 hi .	3
	Abs High	When the absolute value of the signal selected by parameter 33.10 Superv2 act exceeds the value of parameter 33.11 Superv3 hi , bit 2 of 06.13 Superv status is activated. To clear the bit, the absolute value of the signal must fall below the value of parameter 33.12 Superv3 lo .	4
33.10	Superv3 act	Selects the signal to be monitored by supervision 3. See parameter 33.09 Superv3 func.	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	SpRef unramp	03.03 SpeedRef unramp (see page 114).	1073742595
	SpRef ramped	03.05 SpeedRef ramped (see page 114).	1073742597
	SpRef used	03.06 SpeedRef used (see page 114).	1073742598
	TorqRef used	03.14 Torq ref used (see page 115).	1073742606
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-

No.	Name/Value	Description	FbEq
33.11	Superv3 hi	Selects the upper limit for supervision 3. See parameter 33.09 Superv3 func.	
	-32768.00 ... 32768.00	Upper limit for supervision 3.	100 = 1
33.12	Superv3 lo	Selects the lower limit for supervision 3. See parameter 33.09 Superv3 func.	
	-32768.00 ... 32768.00	Lower limit for supervision 3.	100 = 1

34 User load curve		Definition of user load curve. See also section User-definable load curve on page 66.	
34.01	Overload func	Configures the supervision of the upper boundary of the user load curve.	
Bit	Function		
0	Ena sup (Enable supervision) 0 = Disabled: Supervision disabled. 1 = Enabled: Supervision enabled.		
1	Input value sel (Input value selection) 0 = Current: Current is supervised. 1 = Torque: Torque is supervised.		
2	Ena warn (Enable warning) 0 = Disabled 1 = Enabled: Drive generates an alarm when the curve is exceeded.		
3	Ena fault (Enable fault) 0 = Disabled 1 = Enabled: Drive trips on a fault when the curve is exceeded.		
4	Ena lim integ (Enable limit integration) 0 = Disabled 1 = Enabled: Integration time defined by parameter 34.18 Load integ time is used. After the supervision is evoked, the current or torque is limited by the upper boundary of the load curve.		
5	Ena lim always (Enable limit always) 0 = Disabled 1 = Enabled: The current or torque is always limited by the upper boundary of the load curve.		

202 Parameters

No.	Name/Value	Description	FbEq
34.02	Underload func	Configures the supervision of the lower boundary of the user load curve.	
	Bit	Function	
	0	Ena sup (Enable supervision) 0 = Disabled: Supervision disabled. 1 = Enabled: Supervision enabled.	
	1	Input value sel (Input value selection) 0 = Current: Current is supervised. 1 = Torque: Torque is supervised.	
	2	Ena warn (Enable warning) 0 = Disabled 1 = Enabled: Drive generates an alarm when the load remains below the curve for longer than the time defined by parameter 34.20 Underload time .	
	3	Ena fault (Enable fault) 0 = Disabled 1 = Enabled: Drive trips on a fault when the load remains below the curve for longer than the time defined by parameter 34.20 Underload time .	
34.03	Load freq1	Drive output frequency at point 1 of user load curve.	
	1 ... 500 Hz	Frequency at point 1.	1 = 1 Hz
34.04	Load freq2	Drive output frequency at point 2 of user load curve.	
	1 ... 500 Hz	Frequency at point 2.	1 = 1 Hz
34.05	Load freq3	Drive output frequency at point 3 of user load curve.	
	1 ... 500 Hz	Frequency at point 3.	1 = 1 Hz
34.06	Load freq4	Drive output frequency at point 4 of user load curve.	
	1 ... 500 Hz	Frequency at point 4.	1 = 1 Hz
34.07	Load freq5	Drive output frequency at point 5 of user load curve.	
	1 ... 500 Hz	Frequency at point 5.	1 = 1 Hz
34.08	Load low lim1	Minimum load (current or torque) at point 1 of user load curve.	
	0 ... 1600%	Minimum load at point 1.	1 = 1%
34.09	Load low lim2	Minimum load (current or torque) at point 2 of user load curve.	
	0 ... 1600%	Minimum load at point 2.	1 = 1%
34.10	Load low lim3	Minimum load (current or torque) at point 3 of user load curve.	
	0 ... 1600%	Minimum load at point 3.	1 = 1%
34.11	Load low lim4	Minimum load (current or torque) at point 4 of user load curve.	
	0 ... 1600%	Minimum load at point 4.	1 = 1%
34.12	Load low lim5	Minimum load (current or torque) at point 5 of user load curve.	
	0 ... 1600%	Minimum load at point 5.	1 = 1%
34.13	Load high lim1	Maximum load (current or torque) at point 1 of user load curve.	
	0 ... 1600%	Maximum load at point 1.	1 = 1%

No.	Name/Value	Description	FbEq
34.14	Load high lim2	Maximum load (current or torque) at point 2 of user load curve.	
	0 ... 1600%	Maximum load at point 2.	1 = 1%
34.15	Load high lim3	Maximum load (current or torque) at point 3 of user load curve.	
	0 ... 1600%	Maximum load at point 3.	1 = 1%
34.16	Load high lim4	Maximum load (current or torque) at point 4 of user load curve.	
	0 ... 1600%	Maximum load at point 4.	1 = 1%
34.17	Load high lim5	Maximum load (current or torque) at point 5 of user load curve.	
	0 ... 1600%	Maximum load at point 5.	1 = 1%
34.18	Load integ time	Integration time used in limit supervision whenever enabled by parameter 34.01/34.02 .	
	0 ... 10000 s	Integration time.	1 = 1 s
34.19	Load cool time	Defines the cooling time. The output of the overload integrator is set to zero if the load stays continuously below the upper boundary of the user load curve.	
	0 ... 10000 s	Load cooling time.	1 = 1 s
34.20	Underload time	Time for the underload function. See parameter 34.02 Underload func.	
	0 ... 10000 s	Underload time.	1 = 1 s
35 Process variable		Selection and modification of process variables for display as parameters 04.06 ... 04.08 .	
35.01	Signal1 param	Selects a signal to be provided as parameter 04.06 Process var1 .	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	SpRef unram	03.03 SpeedRef unram (see page 114).	1073742595
	SpRef ramped	03.05 SpeedRef ramped (see page 114).	1073742597
	SpRef used	03.06 SpeedRef used (see page 114).	1073742598
	TorqRef used	03.14 Torq ref used (see page 115).	1073742606
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-

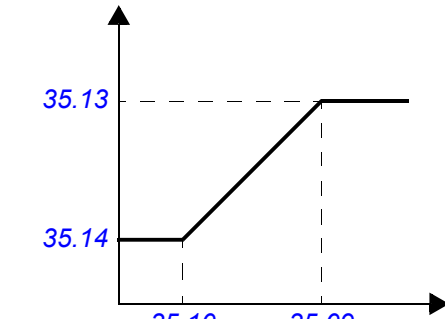
204 Parameters

No.	Name/Value	Description	FbEq
35.02	Signal1 max	<p>Defines the real value of the selected signal that corresponds to the maximum display value defined by parameter 35.06 Proc var1 max.</p> <p><i>04.06 Process var1</i></p> <p>Signal selected by 35.01 Signal1 param</p>	
	-32768...32768	Real signal value corresponding to maximum process variable 1 value.	1 = 1
35.03	Signal1 min	<p>Defines the real value of the selected signal that corresponds to the minimum display value defined by parameter 35.07 Proc var1 min. See diagram at parameter 35.02 Signal1 max.</p>	
	-32768...32768	Real signal value corresponding to minimum process variable 1 value.	1 = 1
35.04	Proc var1 dispf	Scaling for process variable 1. This setting also scales the value for fieldbus.	
	0	1 = 1	0
	1	10 = 1	1
	2	100 = 1	2
	3	1000 = 1	3
	4	10000 = 1	4
	5	100000 = 1	5
35.05	Proc var1 unit	Specifies the unit for parameter 04.06 Process var1 (process variable 1).	
	0	None	0
	1	A	1
	2	V	2
	3	Hz	3
	4	%	4
	5	s	5
	6	h	6
	7	rpm	7
	8	kh	8
	9	C	9
	10	lbft	10
	11	mA	11
	12	mV	12

No.	Name/Value	Description	FbEq
13		kW	13
14		W	14
15		kWh	15
16		F	16
17		hp	17
18		MWh	18
19		m/s	19
20		m ³ /h	20
21		dm ³ /h	21
22		bar	22
23		kPa	23
24		GPM	24
25		PSI	25
26		CFM	26
27		ft	27
28		MGD	28
29		inHg	29
30		FPM	30
31		kbits	31
32		kHz	32
33		Ohm	33
34		ppm	34
35		pps	35
36		l/s	36
37		l/min	37
38		l/h	38
39		m ³ /s	39
40		m ³ /m	40
41		kg/s	41
42		kg/m	42
43		kg/h	43
44		mbar	44
45		Pa	45
46		GPS	46
47		gal/s	47
48		gal/m	48
49		gal/h	49
50		ft ³ /s	50
51		ft ³ /m	51
52		ft ³ /h	52
53		lb/s	53

206 Parameters

No.	Name/Value	Description	FbEq
54		lb/m	54
55		lb/h	55
56		FPS	56
57		ft/s	57
58		inH2O	58
59		inwg	59
60		ftwg	60
61		lbsi	61
62		ms	62
63		Mrev	63
64		days	64
65		inWC	65
66		mpmin	66
67		week	67
68		tonne	68
69		m/s^2	66
70		rev	70
71		deg	71
72		m	72
73		inch	73
74		inc	74
75		m/s^3	75
76		kg/m^2	76
77		kg/m^3	77
78		m^3	78
79		[blank]	79
80		u/s	80
81		u/min	81
82		u/h	82
83...84		[blank]	83...84
85		u/s^2	85
86		min-2	86
87		u/h^2	87
88...89		[blank]	88...89
90		Vrms	90
91		bits	91
92		Nm	92
93		p.u.	93
94		1/s	94
95		mH	95
96		mOhm	96

No.	Name/Value	Description	FbEq
	97	us	97
	98	C/W	98
35.06	Proc var1 max	Maximum value for process variable 1. See diagram at parameter 35.02 Signal1 max .	
	-32768...32768	Maximum value for process variable 1.	1 = 1
35.07	Proc var1 min	Minimum value for process variable 1. See diagram at parameter 35.02 Signal1 max .	
	-32768...32768	Minimum value for process variable 1.	1 = 1
35.08	Signal2 param	Selects a signal to be provided as parameter 04.07 Process var2 .	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	SpRef unramp	03.03 SpeedRef unramp (see page 114).	1073742595
	SpRef ramped	03.05 SpeedRef ramped (see page 114).	1073742597
	SpRef used	03.06 SpeedRef used (see page 114).	1073742598
	TorqRef used	03.14 Torq ref used (see page 115).	1073742606
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
35.09	Signal2 max	Defines the real value of the selected signal that corresponds to the maximum display value defined by parameter 35.13 Proc var2 max . <i>04.07 Process var2</i>  Signal selected by 35.08 Signal2 param	
	-32768...32768	Real signal value corresponding to maximum process variable 2 value.	1 = 1

208 Parameters

No.	Name/Value	Description	FbEq
35.10	Signal2 min	Defines the real value of the selected signal that corresponds to the minimum display value defined by parameter 35.14 Proc var2 min . See diagram at parameter 35.09 Signal2 max .	
	-32768...32768	Real signal value corresponding to minimum process variable 2 value.	1 = 1
35.11	Proc var2 dispf	Scaling for process variable 2. This setting also scales the value for fieldbus.	
	0	1 = 1	0
	1	10 = 1	1
	2	100 = 1	2
	3	1000 = 1	3
	4	10000 = 1	4
	5	100000 = 1	5
35.12	Proc var2 unit	Specifies the unit for parameter 04.07 Process var2 (process variable 2).	
	0...98	See parameter 35.05 Proc var1 unit .	1 = 1
35.13	Proc var2 max	Maximum value for process variable 2. See diagram at parameter 35.09 Signal2 max .	
	-32768...32768	Maximum value for process variable 2.	1 = 1
35.14	Proc var2 min	Minimum value for process variable 2. See diagram at parameter 35.09 Signal2 max .	
	-32768...32768	Minimum value for process variable 2.	1 = 1
35.15	Signal3 param	Selects a signal to be provided as parameter 04.08 Process var3 .	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	SpRef unramp	03.03 SpeedRef unramp (see page 114).	1073742595
	SpRef ramped	03.05 SpeedRef ramped (see page 114).	1073742597
	SpRef used	03.06 SpeedRef used (see page 114).	1073742598
	TorqRef used	03.14 Torq ref used (see page 115).	1073742606
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-

No.	Name/Value	Description	FbEq
35.16	Signal3 max	<p>Defines the real value of the selected signal that corresponds to the maximum display value defined by parameter 35.20 Proc var3 max.</p> <p><i>04.08 Process var3</i></p>	
	-32768...32768	Real signal value corresponding to maximum process variable 3 value.	1 = 1
35.17	Signal3 min	<p>Defines the real value of the selected signal that corresponds to the minimum display value defined by parameter 35.21 Proc var3 min. See diagram at parameter 35.16 Signal3 max.</p>	
	-32768...32768	Real signal value corresponding to minimum process variable 3 value.	1 = 1
35.18	Proc var3 dispf	Scaling for process variable 3. This setting also scales the value for fieldbus.	
	0	1 = 1	0
	1	10 = 1	1
	2	100 = 1	2
	3	1000 = 1	3
	4	10000 = 1	4
	5	100000 = 1	5
35.19	Proc var3 unit	Specifies the unit for parameter <i>04.08 Process var3</i> (process variable 3).	
	0...98	See parameter 35.05 Proc var1 unit .	1 = 1
35.20	Proc var3 max	Maximum value for process variable 3. See diagram at parameter 35.16 Signal3 max .	
	-32768...32768	Maximum value for process variable 3.	1 = 1
35.21	Proc var3 min	Minimum value for process variable 3. See diagram at parameter 35.16 Signal3 max .	
	-32768...32768	Minimum value for process variable 3.	1 = 1
36 Timed functions		Configuration of timers. See also section Timers on page 76 .	
36.01	Timers enable	Enable/disable control for timers. Whenever the source selected by this parameter is off, timers are disabled; when the source is on, timers are enabled.	
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873

210 Parameters

No.	Name/Value	Description	FbEq										
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409										
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945										
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481										
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017										
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947										
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483										
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019										
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-										
	Pointer												
36.02	Timers mode	Specifies whether the time periods defined by parameters 36.03 Start time1 ... 36.18 Stop day4 are valid daily or weekly.											
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Timer1 mode 0 = Daily 1 = Weekly</td> </tr> <tr> <td>1</td> <td>Timer2 mode 0 = Daily 1 = Weekly</td> </tr> <tr> <td>2</td> <td>Timer3 mode 0 = Daily 1 = Weekly</td> </tr> <tr> <td>3</td> <td>Timer4 mode 0 = Daily 1 = Weekly</td> </tr> </tbody> </table>	Bit	Function	0	Timer1 mode 0 = Daily 1 = Weekly	1	Timer2 mode 0 = Daily 1 = Weekly	2	Timer3 mode 0 = Daily 1 = Weekly	3	Timer4 mode 0 = Daily 1 = Weekly	
Bit	Function												
0	Timer1 mode 0 = Daily 1 = Weekly												
1	Timer2 mode 0 = Daily 1 = Weekly												
2	Timer3 mode 0 = Daily 1 = Weekly												
3	Timer4 mode 0 = Daily 1 = Weekly												
36.03	Start time1	Defines the start time for time period 1.											
	00:00:00 ... 24:00:00	Start time for time period 1.	1 = 1 s (24:00:00 = 86400)										
36.04	Stop time1	Defines the stop time for time period 1.											
	00:00:00 ... 24:00:00	Stop time for time period 1.	1 = 1 s (24:00:00 = 86400)										
36.05	Start day1	Defines the week day on which time period 1 begins.											
	Monday	Time period 1 starts on Monday.	1										
	Tuesday	Time period 1 starts on Tuesday.	2										
	Wednesday	Time period 1 starts on Wednesday.	3										
	Thursday	Time period 1 starts on Thursday.	4										
	Friday	Time period 1 starts on Friday.	5										
	Saturday	Time period 1 starts on Saturday.	6										
	Sunday	Time period 1 starts on Sunday.	7										
36.06	Stop day1	Defines the week day on which time period 1 ends.											
	Monday	Time period 1 ends on Monday.	1										
	Tuesday	Time period 1 ends on Tuesday.	2										

No.	Name/Value	Description	FbEq
	Wednesday	Time period 1 ends on Wednesday.	3
	Thursday	Time period 1 ends on Thursday.	4
	Friday	Time period 1 ends on Friday.	5
	Saturday	Time period 1 ends on Saturday.	6
	Sunday	Time period 1 ends on Sunday.	7
36.07	Start time2	Defines the start time for time period 2.	
	00:00:00 ... 24:00:00	Start time for time period 2.	1 = 1 s (24:00:00 = 86400)
36.08	Stop time2	Defines the stop time for time period 2.	
	00:00:00 ... 24:00:00	Stop time for time period 2.	1 = 1 s (24:00:00 = 86400)
36.09	Start day2	Defines the week day on which time period 2 begins.	
	Monday	Time period 2 starts on Monday.	1
	Tuesday	Time period 2 starts on Tuesday.	2
	Wednesday	Time period 2 starts on Wednesday.	3
	Thursday	Time period 2 starts on Thursday.	4
	Friday	Time period 2 starts on Friday.	5
	Saturday	Time period 2 starts on Saturday.	6
	Sunday	Time period 2 starts on Sunday.	7
36.10	Stop day2	Defines the week day on which time period 2 ends.	
	Monday	Time period 2 ends on Monday.	1
	Tuesday	Time period 2 ends on Tuesday.	2
	Wednesday	Time period 2 ends on Wednesday.	3
	Thursday	Time period 2 ends on Thursday.	4
	Friday	Time period 2 ends on Friday.	5
	Saturday	Time period 2 ends on Saturday.	6
	Sunday	Time period 2 ends on Sunday.	7
36.11	Start time3	Defines the start time for time period 3.	
	00:00:00 ... 24:00:00	Start time for time period 3.	1 = 1 s (24:00:00 = 86400)
36.12	Stop time3	Defines the stop time for time period 3.	
	00:00:00 ... 24:00:00	Stop time for time period 3.	1 = 1 s (24:00:00 = 86400)
36.13	Start day3	Defines the week day on which time period 3 begins.	
	Monday	Time period 3 starts on Monday.	1
	Tuesday	Time period 3 starts on Tuesday.	2
	Wednesday	Time period 3 starts on Wednesday.	3
	Thursday	Time period 3 starts on Thursday.	4
	Friday	Time period 3 starts on Friday.	5
	Saturday	Time period 3 starts on Saturday.	6

212 Parameters

No.	Name/Value	Description	FbEq
	Sunday	Time period 3 starts on Sunday.	7
36.14	Stop day3	Defines the week day on which time period 3 ends.	
	Monday	Time period 3 ends on Monday.	1
	Tuesday	Time period 3 ends on Tuesday.	2
	Wednesday	Time period 3 ends on Wednesday.	3
	Thursday	Time period 3 ends on Thursday.	4
	Friday	Time period 3 ends on Friday.	5
	Saturday	Time period 3 ends on Saturday.	6
	Sunday	Time period 3 ends on Sunday.	7
36.15	Start time4	Defines the start time for time period 4.	
	00:00:00 ... 24:00:00	Start time for time period 4.	1 = 1 s (24:00:00 = 86400)
36.16	Stop time4	Defines the stop time for time period 4.	
	00:00:00 ... 24:00:00	Stop time for time period 4.	1 = 1 s (24:00:00 = 86400)
36.17	Start day4	Defines the week day on which time period 4 begins.	
	Monday	Time period 4 starts on Monday.	1
	Tuesday	Time period 4 starts on Tuesday.	2
	Wednesday	Time period 4 starts on Wednesday.	3
	Thursday	Time period 4 starts on Thursday.	4
	Friday	Time period 4 starts on Friday.	5
	Saturday	Time period 4 starts on Saturday.	6
	Sunday	Time period 4 starts on Sunday.	7
36.18	Stop day4	Defines the week day on which time period 4 ends.	
	Monday	Time period 4 ends on Monday.	1
	Tuesday	Time period 4 ends on Tuesday.	2
	Wednesday	Time period 4 ends on Wednesday.	3
	Thursday	Time period 4 ends on Thursday.	4
	Friday	Time period 4 ends on Friday.	5
	Saturday	Time period 4 ends on Saturday.	6
	Sunday	Time period 4 ends on Sunday.	7
36.19	Boost signal	Boosting can be used to extend the timer enable signal for the time defined by parameter 36.20 Boost time . The boost time starts when the boost signal changes state from 1 to 0.	
	DI1	Digital input DI1 (as indicated by 02.01 DI status , bit 0).	1073742337
	DI2	Digital input DI2 (as indicated by 02.01 DI status , bit 1).	1073807873
	DI3	Digital input DI3 (as indicated by 02.01 DI status , bit 2).	1073873409
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017

No.	Name/Value	Description	FbEq												
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947												
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483												
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019												
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-												
	Pointer														
36.20	Boost time	Boost time. See parameter 36.19 Boost signal .													
	00:00:00 ... 24:00:00	Boost time.	1 = 1 s (24:00:00 = 86400)												
36.21	Timed func1	<p>Selects which time periods (1...4) are used with timed function 1. Also determines whether boost is used with timed function 1.</p> <p>The parameter is a 16-bit word with each bit corresponding to a function. Whenever a bit is set to 1, the corresponding function is in use.</p> <p>The bits of the binary number correspond to the following functions:</p>													
<table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Timer1 ena (Time period 1 enable)</td> </tr> <tr> <td>1</td> <td>Timer2 ena (Time period 2 enable)</td> </tr> <tr> <td>2</td> <td>Timer3 ena (Time period 3 enable)</td> </tr> <tr> <td>3</td> <td>Timer4 ena (Time period 4 enable)</td> </tr> <tr> <td>4</td> <td>Boost ena (Boost enable)</td> </tr> </tbody> </table>				Bit	Function	0	Timer1 ena (Time period 1 enable)	1	Timer2 ena (Time period 2 enable)	2	Timer3 ena (Time period 3 enable)	3	Timer4 ena (Time period 4 enable)	4	Boost ena (Boost enable)
Bit	Function														
0	Timer1 ena (Time period 1 enable)														
1	Timer2 ena (Time period 2 enable)														
2	Timer3 ena (Time period 3 enable)														
3	Timer4 ena (Time period 4 enable)														
4	Boost ena (Boost enable)														
36.22	Timed func2	<p>Selects which time periods (1...4) are used with timed function 2. Also determines whether boost is used with timed function 2.</p> <p>The parameter is a 16-bit word with each bit corresponding to a function. Whenever a bit is set to 1, the corresponding function is in use.</p> <p>The bits of the binary number correspond to the following functions:</p>													
<table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Timer1 ena (Time period 1 enable)</td> </tr> <tr> <td>1</td> <td>Timer2 ena (Time period 2 enable)</td> </tr> <tr> <td>2</td> <td>Timer3 ena (Time period 3 enable)</td> </tr> <tr> <td>3</td> <td>Timer4 ena (Time period 4 enable)</td> </tr> <tr> <td>4</td> <td>Boost ena (Boost enable)</td> </tr> </tbody> </table>				Bit	Function	0	Timer1 ena (Time period 1 enable)	1	Timer2 ena (Time period 2 enable)	2	Timer3 ena (Time period 3 enable)	3	Timer4 ena (Time period 4 enable)	4	Boost ena (Boost enable)
Bit	Function														
0	Timer1 ena (Time period 1 enable)														
1	Timer2 ena (Time period 2 enable)														
2	Timer3 ena (Time period 3 enable)														
3	Timer4 ena (Time period 4 enable)														
4	Boost ena (Boost enable)														

No.	Name/Value	Description	FbEq												
36.23	Timed func3	<p>Selects which time periods (1...4) are used with timed function 3. Also determines whether boost is used with timed function 3.</p> <p>The parameter is a 16-bit word with each bit corresponding to a function. Whenever a bit is set to 1, the corresponding function is in use.</p> <p>The bits of the binary number correspond to the following functions:</p>													
<table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Timer1 ena (Time period 1 enable)</td> </tr> <tr> <td>1</td> <td>Timer2 ena (Time period 2 enable)</td> </tr> <tr> <td>2</td> <td>Timer3 ena (Time period 3 enable)</td> </tr> <tr> <td>3</td> <td>Timer4 ena (Time period 4 enable)</td> </tr> <tr> <td>4</td> <td>Boost ena (Boost enable)</td> </tr> </tbody> </table>				Bit	Function	0	Timer1 ena (Time period 1 enable)	1	Timer2 ena (Time period 2 enable)	2	Timer3 ena (Time period 3 enable)	3	Timer4 ena (Time period 4 enable)	4	Boost ena (Boost enable)
Bit	Function														
0	Timer1 ena (Time period 1 enable)														
1	Timer2 ena (Time period 2 enable)														
2	Timer3 ena (Time period 3 enable)														
3	Timer4 ena (Time period 4 enable)														
4	Boost ena (Boost enable)														
36.24	Timed func4	<p>Selects which time periods (1...4) are used with timed function 4. Also determines whether boost is used with timed function 4.</p> <p>The parameter is a 16-bit word with each bit corresponding to a function. Whenever a bit is set to 1, the corresponding function is in use.</p> <p>The bits of the binary number correspond to the following functions:</p>													
<table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Timer1 ena (Time period 1 enable)</td> </tr> <tr> <td>1</td> <td>Timer2 ena (Time period 2 enable)</td> </tr> <tr> <td>2</td> <td>Timer3 ena (Time period 3 enable)</td> </tr> <tr> <td>3</td> <td>Timer4 ena (Time period 4 enable)</td> </tr> <tr> <td>4</td> <td>Boost ena (Boost enable)</td> </tr> </tbody> </table>				Bit	Function	0	Timer1 ena (Time period 1 enable)	1	Timer2 ena (Time period 2 enable)	2	Timer3 ena (Time period 3 enable)	3	Timer4 ena (Time period 4 enable)	4	Boost ena (Boost enable)
Bit	Function														
0	Timer1 ena (Time period 1 enable)														
1	Timer2 ena (Time period 2 enable)														
2	Timer3 ena (Time period 3 enable)														
3	Timer4 ena (Time period 4 enable)														
4	Boost ena (Boost enable)														

38 Flux ref	Flux reference and <i>U/f</i> curve settings. See also section <i>User-definable U/f curve</i> on page 67.	
38.01 Flux ref	Sets the flux reference (in percent of parameter <i>99.08 Mot nom freq</i>) at field weakening point.	
0 ... 200%	Flux reference at field weakening point.	1 = 1%
38.03 <i>U/f</i> curve func	<p>Selects the form of the <i>U/f</i> (voltage/frequency) curve below the field weakening point.</p> <p>Note: This functionality can be used in scalar control only, i.e. when <i>99.05 Motor ctrl mode</i> setting is <i>Scalar</i>.</p>	
Linear	Linear <i>U/f</i> curve. Recommended for constant-torque applications.	0
Quadratic	Quadratic <i>U/f</i> curve. Recommended for centrifugal pump and fan applications.	1
User	Custom <i>U/f</i> curve. The curve is formed by the points defined by parameters <i>38.04...38.13</i> .	2
38.04 <i>U/f</i> curve freq1	Defines the frequency at the 1st point on the custom <i>U/f</i> curve in percent of parameter <i>99.08 Mot nom freq</i> . In use when <i>38.03 U/f curve func</i> is set to <i>User</i> .	
1 ... 500%	1st point, frequency.	1 = 1%
38.05 <i>U/f</i> curve freq2	Defines the frequency at the 2nd point on the custom <i>U/f</i> curve in percent of parameter <i>99.08 Mot nom freq</i> .	

No.	Name/Value	Description	FbEq
	1 ... 500%	2nd point, frequency.	1 = 1%
38.06	U/f curve freq3	Defines the frequency at the 3rd point on the custom <i>U/f</i> curve in percent of parameter 99.08 Mot nom freq.	
	1 ... 500%	3rd point, frequency.	1 = 1%
38.07	U/f curve freq4	Defines the frequency at the 4th point on the custom <i>U/f</i> curve in percent of parameter 99.08 Mot nom freq.	
	1 ... 500%	4th point, frequency.	1 = 1%
38.08	U/f curve freq5	Defines the frequency at the 5th point on the custom <i>U/f</i> curve in percent of parameter 99.08 Mot nom freq.	
	1 ... 500%	5th point, frequency.	1 = 1%
38.09	U/f curve volt1	Defines the voltage at the 1st point on the custom <i>U/f</i> curve in percent of parameter 99.07 Mot nom voltage.	
	0 ... 200%	1st point, voltage.	1 = 1%
38.10	U/f curve volt2	Defines the voltage at the 2nd point on the custom <i>U/f</i> curve in percent of parameter 99.07 Mot nom voltage.	
	0 ... 200%	2nd point, voltage.	1 = 1%
38.11	U/f curve volt3	Defines the voltage at the 3rd point on the custom <i>U/f</i> curve in percent of parameter 99.07 Mot nom voltage.	
	0 ... 200%	3rd point, voltage.	1 = 1%
38.12	U/f curve volt4	Defines the voltage at the 4th point on the custom <i>U/f</i> curve in percent of parameter 99.07 Mot nom voltage.	
	0 ... 200%	4th point, voltage.	1 = 1%
38.13	U/f curve volt5	Defines the voltage at the 5th point on the custom <i>U/f</i> curve in percent of parameter 99.07 Mot nom voltage.	
	0 ... 200%	5th point, voltage.	1 = 1%
38.16	Flux ref pointer	Selects the source of the flux reference.	
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-

40 Motor control		Motor control settings.	
40.01	Motor noise	An optimization setting for balancing between control performance and motor noise level.	
	Cyclic	Control performance optimized for cyclic load applications. Note: With this setting, the maximum motor cable length is smaller than with Default .	0
	Low noise	Minimizes motor noise; control performance optimized for high (> 300 Hz) output frequencies. Note: Drive loadability is reduced with this setting and some derating must be applied if a certain constant output current is needed. This setting is not recommended for cyclic load applications. The maximum motor cable length is 50 m (164 ft) with drives up to 45 kW.	1
	Default	Control performance optimized for long motor cables.	2

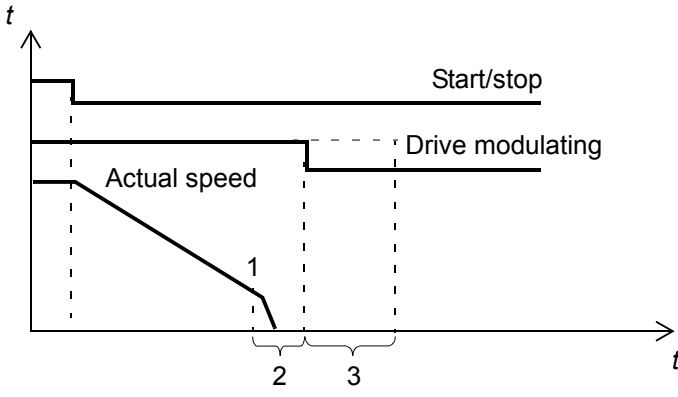
No.	Name/Value	Description	FbEq
40.03	Slip gain	<p>Defines the slip gain which is used to improve the estimated motor slip. 100% means full slip gain; 0% means no slip gain. The default value is 100%. Other values can be used if a static speed error is detected despite of the full slip gain.</p> <p>Example (with nominal load and nominal slip of 40 rpm): A 1000 rpm constant speed reference is given to the drive. Despite of the full slip gain (= 100%), a manual tachometer measurement from the motor axis gives a speed value of 998 rpm. The static speed error is 1000 rpm - 998 rpm = 2 rpm. To compensate the error, the slip gain should be increased. At the 105% gain value, no static speed error exists (2 rpm / 40 rpm = 5%).</p>	
	0 ... 200%	Slip gain.	1 = 1%
40.04	Voltage reserve	<p>Defines the minimum allowed voltage reserve. When the voltage reserve has decreased to the set value, the drive enters the field weakening area.</p> <p>If the intermediate circuit DC voltage $U_{dc} = 550 \text{ V}$ and the voltage reserve is 5%, the RMS value of the maximum output voltage in steady-state operation is</p> $0.95 \times 550 \text{ V} / \sqrt{2} = 369 \text{ V}$ <p>The dynamic performance of the motor control in the field weakening area can be improved by increasing the voltage reserve value, but the drive enters the field weakening area earlier.</p>	
	-4 ... 50%	Voltage reserve.	1 = 1%
40.06	Force open loop	Defines the speed/position information used by the motor model.	
	False	Motor model uses the speed feedback selected by parameter 19.02 Speed fb sel .	0
	True	Motor model uses the internal speed estimate (even when parameter 19.02 Speed fb sel is set to Enc1 speed / Enc2 speed).	1

No.	Name/Value	Description	FbEq
40.07	IR-compensation	<p>Defines the relative output voltage boost at zero speed (IR compensation). The function is useful in applications with a high break-away torque where direct torque control (DTC mode) cannot be applied.</p> <p>See also section IR compensation for a scalar controlled drive on page 66.</p>	
	0.00 ... 50.00%	Voltage boost at zero speed in percent of nominal motor voltage.	100 = 1%
40.10	Flux braking	Defines the level of braking power.	
	Disabled	Flux braking is disabled.	0
	Moderate	Flux level is limited during the braking. Deceleration time is longer compared to full braking.	1
	Full	Maximum braking power. Almost all available current is used to convert the mechanical braking energy to thermal energy in the motor.	2
42 Mech brake ctrl		Mechanical brake control configuration. See also section Mechanical brake control on page 72.	
42.01	Brake ctrl	<p>Activates the brake control function with or without supervision.</p> <p>Note: This parameter cannot be changed while the drive is running.</p>	
	No	Brake control disabled.	0
	With ack	Brake control enabled with supervision (supervision is activated by parameter 42.02 Brake acknowl).	1
	No ack	Brake control enabled without supervision.	2

No.	Name/Value	Description	FbEq
42.02	Brake acknowl	<p>Selects the source for the external brake on/off supervision activation (when parameter 42.01 Brake ctrl is set to <i>With ack</i>). The use of the external on/off supervision signal is optional.</p> <p>1 = The brake is open 0 = The brake is closed</p> <p>Brake supervision is usually controlled through a digital input. When a brake control error is detected, the drive reacts as defined by parameter 42.12 Brake fault func.</p> <p>Note: This parameter cannot be changed while the drive is running.</p>	
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
42.03	Open delay	<p>Defines the brake open delay (= the delay between the internal open brake command and the release of the motor speed control). The delay counter starts when the drive has magnetised the motor and risen the motor torque to the level required at the brake release (parameter 42.08 Brake open torq). Simultaneously with the counter start, the brake function energises the relay output controlling the brake and the brake starts opening.</p> <p>Set the delay the same as the mechanical opening delay of the brake specified by the brake manufacturer.</p>	
	0.00 ... 5.00 s	Brake open delay.	100 = 1 s
42.04	Close delay	<p>Defines the brake close delay. The delay counter starts when the motor actual speed has fallen below the set level (parameter 42.05 Close speed) after the drive has received the stop command. Simultaneously with the counter start, the brake control function de-energises the relay output controlling the brake and the brake starts closing. During the delay, the brake function keeps the motor live preventing the motor speed from falling below zero.</p> <p>Set the delay time to the same value as the mechanical make-up time of the brake (= operating delay when closing) specified by the brake manufacturer.</p>	
	0.00 ... 60.00 s	Brake close delay.	100 = 1 s
42.05	Close speed	Defines the brake close speed (as an absolute value). See parameter 42.04 Close delay .	
	0.0 ... 1000.0 rpm	Brake close speed.	10 = 1 rpm
42.06	Close cmd delay	Defines a close command delay, i.e. the time between when brake close conditions are met and when the close command is given.	
	0.00 ... 10.00 s	Brake close command delay.	100 = 1 s

No.	Name/Value	Description	FbEq
42.07	Reopen delay	Defines a reopen delay, i.e. the time between when the close command is given and when the brake can be reopened.	
	0.00 ... 10.00 s	Brake reopen delay.	100 = 1 s
42.08	Brake open torq	Defines the motor starting torque at brake release (in percent of the motor nominal torque) when parameter 42.09 Open torq src is set to P.42.08 . Note: If different from 0, this value overrides the setting of parameter 42.09 Open torq src .	
	-1000.0 ... 1000.0%	Motor starting torque at brake release.	10 = 1%
42.09	Open torq src	Selects the source for the “brake open” torque value (motor starting torque at brake release). See also parameter 42.08 Brake open torq .	
	Zero	Zero speed reference.	0
	AI1 scaled	02.05 AI1 scaled (see page 105).	1073742341
	AI2 scaled	02.07 AI2 scaled (see page 105).	1073742343
	FBA ref1	02.26 FBA main ref1 (see page 109).	1073742362
	FBA ref2	02.27 FBA main ref2 (see page 109).	1073742363
	D2D ref1	02.32 D2D ref1 (see page 110).	1073742368
	D2D ref2	02.33 D2D ref2 (see page 110).	1073742369
	Brk torq mem	03.15 Brake torq mem (see page 115).	1073742607
	P.42.08	Parameter 42.08 Brake open torq .	1073752584
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
42.10	Brake close req	Selects the source for the brake close/open request. 1 = Brake close request 0 = Brake open request Note: This parameter cannot be changed while the drive is running.	
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
42.11	Brake hold open	Selects the source for the activation of the brake open command hold. 1 = Hold active 0 = Normal operation Note: This parameter cannot be changed while the drive is running.	
	DI4	Digital input DI4 (as indicated by 02.01 DI status , bit 3).	1073938945
	DI5	Digital input DI5 (as indicated by 02.01 DI status , bit 4).	1074004481

No.	Name/Value	Description	FbEq
	DI6	Digital input DI6 (as indicated by 02.01 DI status , bit 5).	1074070017
	DIO4	Digital input/output DIO4 (as indicated by 02.03 DIO status , bit 3).	1073938947
	DIO5	Digital input/output DIO5 (as indicated by 02.03 DIO status , bit 4).	1074004483
	DIO6	Digital input/output DIO6 (as indicated by 02.03 DIO status , bit 5).	1074070019
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
42.12	Brake fault func	Defines how the drive reacts in case of mechanical brake control error. If brake control supervision has not been activated by parameter 42.01 Brake ctrl , this parameter is disabled.	
	Fault	The drive trips on fault BRAKE NOT CLOSED / BRAKE NOT OPEN if the status of the optional external brake acknowledgement signal does not meet the status presumed by the brake control function. The drive trips on fault BRAKE START TORQUE if the required motor starting torque at brake release is not achieved.	0
	Alarm	The drive generates alarm BRAKE NOT CLOSED / BRAKE NOT OPEN if the status of the optional external brake acknowledgement signal does not meet the status presumed by the brake control function. The drive generates alarm BRAKE START TORQUE if the required motor starting torque at brake release is not achieved.	1
	Open flt	The drive generates alarm BRAKE NOT CLOSED (when closing the brake) and trips on fault BRAKE NOT OPEN (when opening the brake) if the status of the optional external brake acknowledgement signal does not match the status presumed by the brake control logic. The drive trips on BRAKE START TORQUE if the required motor start torque at brake release is not achieved.	2
42.13	Close flt delay	Defines a close fault delay, i.e. the time between when the brake is closed and when a brake close fault is generated.	
	0.00 ... 600.00 s	Brake close fault delay.	100 = 1 s

No.	Name/Value	Description	FbEq						
42.14	Extend run time	<p>Defines an extended run time for the brake control function at stop. During the delay, the motor is kept magnetized (modulating) and ready for an immediate restart.</p> <p>0.0 s = Normal stop routine of the brake control function: The motor magnetization (modulation) is switched off after the brake close delay has passed.</p> <p>0.1...3600.0 s = Extended stop routine of the brake control function: The motor magnetization (modulation) is switched off after the brake close delay and the extended run time have passed. During the extended run time, a zero torque reference is applied, and the motor is ready for an immediate restart.</p>  <p>1 = Brake close speed 2 = Brake close delay 3 = Extended run time</p>							
	0.0 ... 3600.0 s	Extended run time.	100 = 1 s						
44 Maintenance		Maintenance counter configuration. See also section Maintenance counters on page 84.							
44.01	Ontime1 func	<p>Configures on-time counter 1. This counter runs whenever the signal selected by parameter 44.02 Ontime1 src is on. After the limit set by parameter 44.03 Ontime1 limit is reached, an alarm specified by parameter 44.04 Ontime1 alm sel is given, and the counter reset.</p> <p>The current value of the counter is readable from parameter 04.09 Counter ontime1. Bit 0 of 06.15 Counter status indicates that the count has exceeded the limit.</p> <table border="1" data-bbox="294 1601 1430 1848"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.</td> </tr> <tr> <td>1</td> <td>Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.</td> </tr> </tbody> </table>	Bit	Function	0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.	1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.	
Bit	Function								
0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.								
1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.								
44.02	Ontime1 src	Selects the signal to be monitored by on-time counter 1. See parameter 44.01 Ontime1 func .							
	RO1	Relay output RO1 (as indicated by 02.02 RO status , bit 0).	1073742338						
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969						
	Charged	Bit 9 of 06.02 Status word2 (see page 117).	1074333186						

222 Parameters

No.	Name/Value	Description	FbEq						
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-						
	Pointer								
44.03	Ontime1 limit	Sets the alarm limit for on-time counter 1. See parameter 44.01 Ontime1 func.							
	0...2147483647 s	Alarm limit for on-time counter 1.							
44.04	Ontime1 alm sel	Selects the alarm for on-time counter 1. See parameter 44.01 Ontime1 func.							
	On-time1	Pre-selectable alarm for on-time counter 1.	0						
	Device clean	Pre-selectable alarm for on-time counter 1.	1						
	Add cool fan	Pre-selectable alarm for on-time counter 1.	2						
	Cabinet fan	Pre-selectable alarm for on-time counter 1.	3						
	Dc-capacitor	Pre-selectable alarm for on-time counter 1.	4						
	Mot bearing	Pre-selectable alarm for on-time counter 1.	5						
44.05	Ontime2 func	Configures on-time counter 2. This counter runs whenever the signal selected by parameter 44.06 Ontime2 src is on. After the limit set by parameter 44.07 Ontime2 limit is reached, an alarm specified by parameter 44.08 Ontime2 alm sel is given, and the counter reset. The current value of the counter is readable from parameter 04.10 Counter ontime2 . Bit 1 of 06.15 Counter status indicates that the count has exceeded the limit.							
<table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.</td> </tr> <tr> <td>1</td> <td>Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.</td> </tr> </tbody> </table>				Bit	Function	0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.	1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.
Bit	Function								
0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.								
1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.								
44.06	Ontime2 src	Selects the signal to be monitored by on-time counter 2. See parameter 44.05 Ontime2 func.							
	RO1	Relay output RO1 (as indicated by 02.02 RO status , bit 0).	1073742338						
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969						
	Charged	Bit 9 of 06.02 Status word2 (see page 117).	1074333186						
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-						
	Pointer								
44.07	Ontime2 limit	Sets the alarm limit for on-time counter 2. See parameter 44.05 Ontime2 func.							
	0 ... 2147483647 s	Alarm limit for on-time counter 2.	1 = 1 s						
44.08	Ontime2 alm sel	Selects the alarm for on-time counter 2. See parameter 44.05 Ontime2 func.							
	On-time2	Pre-selectable alarm for on-time counter 2.	0						
	Device clean	Pre-selectable alarm for on-time counter 2.	1						
	Add cool fan	Pre-selectable alarm for on-time counter 2.	2						
	Cabinet fan	Pre-selectable alarm for on-time counter 2.	3						
	Dc-capacitor	Pre-selectable alarm for on-time counter 2.	4						
	Mot bearing	Pre-selectable alarm for on-time counter 2.	5						

No.	Name/Value	Description	FbEq						
44.09	Edge count1 func	<p>Configures rising edge counter 1. This counter is incremented every time the signal selected by parameter 44.10 Edge count1 src switches on (unless a divisor value is applied – see parameter 44.12 Edge count1 div). After the limit set by parameter 44.11 Edge count1 lim is reached, an alarm specified by parameter 44.13 Edg cnt1 alm sel is given, and the counter reset.</p> <p>The current value of the counter is readable from parameter 04.11 Counter edge1. Bit 2 of 06.15 Counter status indicates that the count has exceeded the limit.</p>							
<table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td> Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset. </td> </tr> <tr> <td>1</td> <td> Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached. </td> </tr> </tbody> </table>				Bit	Function	0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.	1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.
Bit	Function								
0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.								
1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.								
44.10	Edge count1 src	Selects the signal to be monitored by rising edge counter 1. See parameter 44.09 Edge count1 func .							
	RO1	Relay output RO1 (as indicated by 02.02 RO status , bit 0).	1073742338						
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969						
	Charged	Bit 9 of 06.02 Status word2 (see page 117).	1074333186						
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-						
	Pointer								
44.11	Edge count1 lim	Sets the alarm limit for rising edge counter 1. See parameter 44.09 Edge count1 func .							
	0 ... 2147483647	Alarm limit for rising edge counter 1.	1 = 1						
44.12	Edge count1 div	Divisor for rising edge counter 1. Determines how many rising edges increment the counter by 1.							
	1 ... 2147483647	Divisor for rising edge counter 1.	1 = 1						
44.13	Edg cnt1 alm sel	Selects the alarm for rising edge counter 1. See parameter 44.09 Edge count1 func .							
	Edge count1	Pre-selectable alarm for rising edge counter 1.	0						
	Main cntactr	Pre-selectable alarm for rising edge counter 1.	1						
	Output relay	Pre-selectable alarm for rising edge counter 1.	2						
	Motor starts	Pre-selectable alarm for rising edge counter 1.	3						
	Power ups	Pre-selectable alarm for rising edge counter 1.	4						
	Dc-charge	Pre-selectable alarm for rising edge counter 1.	5						

No.	Name/Value	Description	FbEq						
44.14	Edge count2 func	<p>Configures rising edge counter 2. The counter is incremented every time the signal selected by parameter 44.15 Edge count2 src switches on (unless a divisor value is applied – see parameter 44.17 Edge count2 div). After the limit set by parameter 44.16 Edge count2 lim is reached, an alarm specified by parameter 44.22 Edg cnt2 alm sel is given and the counter is reset.</p> <p>The current value of the counter is readable from parameter 04.12 Counter edge2. Bit 3 of 06.15 Counter status indicates that the count has exceeded the limit.</p>							
<table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td> Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset. </td> </tr> <tr> <td>1</td> <td> Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached. </td> </tr> </tbody> </table>				Bit	Function	0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.	1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.
Bit	Function								
0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.								
1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.								
44.15	Edge count2 src	Selects the signal to be monitored by rising edge counter 2. See parameter 44.14 Edge count2 func .							
	RO1	Relay output RO1 (as indicated by 02.02 RO status , bit 0).	1073742338						
	Running	Bit 3 of 06.01 Status word1 (see page 116).	1073939969						
	Charged	Bit 9 of 06.02 Status word2 (see page 117).	1074333186						
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-						
	Pointer								
44.16	Edge count2 lim	Sets the alarm limit for rising edge counter 2. See parameter 44.14 Edge count2 func .							
	0 ... 2147483647	Alarm limit for rising edge counter 2.	1 = 1						
44.17	Edge count2 div	Divisor for rising edge counter 2. Determines how many rising edges increment the counter by 1.							
	1 ... 2147483647	Divisor for rising edge counter 2.	1 = 1						
44.18	Edg cnt2 alm sel	Selects the alarm for rising edge counter 2. See parameter 44.14 Edge count2 func .							
	Edge count2	Pre-selectable alarm for rising edge counter 2.	0						
	Main cntactr	Pre-selectable alarm for rising edge counter 2.	1						
	Output relay	Pre-selectable alarm for rising edge counter 2.	2						
	Motor starts	Pre-selectable alarm for rising edge counter 2.	3						
	Power ups	Pre-selectable alarm for rising edge counter 2.	4						
	Dc-charge	Pre-selectable alarm for rising edge counter 2.	5						



No.	Name/Value	Description	FbEq						
44.19	Val count1 func	<p>Configures value counter 1. This counter measures, by integration, the area below the signal selected by parameter 44.20 Val count1 src. When the total area exceeds the limit set by parameter 44.21 Val count1 lim, an alarm is given (if enabled by bit 1 of this parameter).</p> <p>The signal is sampled at 1-second intervals. Note that the scaled (see the “FbEq” column at the signal in question) value of the signal is used.</p> <p>The current value of the counter is readable from parameter 04.13 Counter value1. Bit 4 of 06.15 Counter status indicates that the counter has exceeded the limit.</p>							
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td> Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset. </td> </tr> <tr> <td>1</td> <td> Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached. </td> </tr> </tbody> </table>	Bit	Function	0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.	1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.	
Bit	Function								
0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.								
1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.								
44.20	Val count1 src	Selects the signal to be monitored by value counter 1. See parameter 44.19 Val count1 func .							
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081						
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-						
44.21	Val count1 lim	Sets the alarm limit for value counter 1. See parameter 44.19 Val count1 func .							
	0 ... 2147483647	Alarm limit for value counter 1.	1 = 1						
44.22	Val count1 div	Divisor for value counter 1. The value of the monitored signal is divided by this value before integration.							
	1 ... 2147483647	Divisor for value counter 1.	1 = 1						
44.23	Val cnt1 alm sel	Selects the alarm for value counter 1. See parameter 44.19 Val count1 func .							
	Value1	Pre-selectable alarm for value counter 1.	0						
	Mot bearing	Pre-selectable alarm for value counter 1.	1						

No.	Name/Value	Description	FbEq						
44.24	Val count2 func	<p>Configures value counter 2. This counter measures, by integration, the area below the signal selected by parameter 44.25 Val count2 src. When the total area exceeds the limit set by parameter 44.26 Val count2 lim, an alarm is given (if enabled by bit 1 of this parameter).</p> <p>The signal is sampled at 1-second intervals. Note that the scaled (see the “FbEq” column at the signal in question) value of the signal is used.</p> <p>The current value of the counter is readable from parameter 04.14 Counter value2. Bit 5 of 06.15 Counter status indicates that the counter has exceeded the limit.</p>							
<table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td> Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset. </td> </tr> <tr> <td>1</td> <td> Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached. </td> </tr> </tbody> </table>				Bit	Function	0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.	1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.
Bit	Function								
0	Counter mode 0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds. 1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset.								
1	Alarm ena (Alarm enable) 0 = Disable: No alarm is given when limit is reached. 1 = Enable: Alarm is given when limit is reached.								
44.25	Val count2 src	Selects the signal to be monitored by value counter 2. See parameter 44.24 Val count2 func .							
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081						
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-						
44.26	Val count2 lim	Sets the alarm limit for value counter 2. See parameter 44.24 Val count2 func .							
	0 ... 2147483647	Alarm limit for value counter 2.	1 = 1						
44.27	Val count2 div	Divisor for value counter 2. The value of the monitored signal is divided by this value before integration.							
	1 ... 2147483647	Divisor for value counter 2.	1 = 1						
44.28	Val cnt2 alm sel	Selects the alarm for value counter 2. See parameter 44.24 Val count2 func .							
	Value2	Pre-selectable alarm for value counter 2.	0						
	Mot bearing	Pre-selectable alarm for value counter 2.	1						
44.29	Fan ontime lim	Sets the limit for the cooling fan on-time counter. The counter monitors signal 01.28 Fan on-time (see page 104). When the signal reaches the limit, alarm 2056 COOLING FAN (0x5081) is given.							
	0.00 ... 35791394.11 h	Alarm limit for cooling fan on-time.	1 = 1 min						
44.30	Runtime lim	Sets the limit for the drive run-time counter. The counter monitors signal 01.27 Run-time counter (see page 104). When the signal reaches the limit, the alarm specified by parameter 44.31 Runtime alm sel is given.							
	0.00 ... 35791394.11 h	Alarm limit for the drive run-time counter.	1 = 1 min						
44.31	Runtime alm sel	Selects the alarm for the drive run time counter.							
	Device clean	Pre-selectable alarm for the drive run time counter.	1						
	Add cool fan	Pre-selectable alarm for the drive run time counter.	2						
	Cabinet fan	Pre-selectable alarm for the drive run time counter.	3						

No.	Name/Value	Description	FbEq
	Dc-capacitor	Pre-selectable alarm for the drive run time counter.	4
	Mot bearing	Pre-selectable alarm for the drive run time counter.	5
44.32	kWh inv lim	Sets the limit for the energy counter. The counter monitors signal 01.24 kWh inverter (see page 103). When the signal reaches the limit, the alarm specified by parameter 44.33 kWh inv alm sel is given.	
	0 ... 2147483647	Alarm limit for the energy counter.	1 = 1 kWh
44.33	kWh inv alm sel	Selects the alarm for the energy counter.	
	Device clean	Pre-selectable alarm for the energy counter.	1
	Add cool fan	Pre-selectable alarm for the energy counter.	2
	Cabinet fan	Pre-selectable alarm for the energy counter.	3
	Dc-capacitor	Pre-selectable alarm for the energy counter.	4
	Mot bearing	Pre-selectable alarm for the energy counter.	5
45 Energy optimising		Energy optimization settings. See also section Energy saving calculator on page 85.	
45.01	Energy optim	Enables/disables energy optimization function. The function optimizes the flux so that total energy consumption and motor noise level are reduced when the drive operates below the nominal load. The total efficiency (motor and drive) can be improved by 1...10% depending on load torque and speed. Note: With a permanent magnet motor, energy optimization is always enabled regardless of this parameter.	
	Disable	Energy optimization disabled.	0
	Enable	Energy optimization enabled.	1
45.02	Energy tariff1	Price of energy per kWh. Used for reference when savings are calculated. See parameters 01.35 Saved energy , 01.36 Saved amount and 01.37 Saved CO2 .	
	0.00 ... 21474836.47	Price of energy per kWh.	1 = 1
45.06	E tariff unit	Specifies the currency used for the savings calculation.	
	Local	The currency is determined by the setting of parameter 99.01 Language .	0
	Eur	Euro.	1
	Usd	US dollar.	2
45.07	CO2 Conv factor	Conversion factor for converting energy into CO2 emissions (kg/kWh or tn/MWh). Used for multiplying the saved energy in MWh to calculate the value of signal 01.37 Saved CO2 (reduction on carbon dioxide emissions in metric tons). $01.37 \text{ Saved CO2} = 01.35 \text{ Saved energy (MWh)} \times 45.07 \text{ CO2 Conv factor (tn/MWh)}$	
	0.0...10.0	Conversion factor for converting energy into CO2 emissions (kg/kWh or tn/MWh).	1 = 1
45.08	Reference power	Motor power when connected directly to supply. Used for reference when energy savings are calculated. See parameters 01.35 Saved energy , 01.36 Saved amount and 01.37 Saved CO2 . Note: The accuracy of the energy savings calculation is directly dependent on the accuracy of this value.	
	00.0... 1000.0%	Motor power in percent of nominal motor power.	1 = 1

No.	Name/Value	Description	FbEq
45.09	Energy reset	Resets the energy counters 01.35 Saved energy , 01.36 Saved amount and 01.37 Saved CO2 .	
	Done	Reset not requested (normal operation).	0
	Reset	Reset energy counters. The value reverts automatically to Done .	1
47 Voltage ctrl		Overvoltage and undervoltage control settings. See also section DC voltage control on page 77 .	
47.01	Overvolt ctrl	Enables the overvoltage control of the intermediate DC link. Fast braking of a high inertia load causes the voltage to rise to the overvoltage control limit. To prevent the DC voltage from exceeding the limit, the overvoltage controller automatically decreases the braking torque. Note: If a brake chopper and resistor or a regenerative supply section are included in the drive, the controller must be disabled.	
	Disable	Overvoltage control disabled.	0
	Enable	Overvoltage control enabled.	1
47.02	Undervolt ctrl	Enables the undervoltage control of the intermediate DC link. If the DC voltage drops due to input power cut off, the undervoltage controller will automatically decrease the motor torque in order to keep the voltage above the lower limit. By decreasing the motor torque, the inertia of the load will cause regeneration back to the drive, keeping the DC link charged and preventing an undervoltage trip until the motor coasts to stop. This will act as a power-loss ride-through functionality in systems with high inertia, such as a centrifuge or a fan.	
	Disable	Undervoltage control disabled.	0
	Enable	Undervoltage control enabled.	1
47.03	SupplyVoltAutold	Enables the auto-identification of the supply voltage.	
	Disable	Auto-identification of supply voltage disabled.	0
	Enable	Auto-identification of supply voltage enabled.	1
47.04	Supply voltage	Defines the nominal supply voltage. Used if auto-identification of the supply voltage is not enabled by parameter 47.03 SupplyVoltAutold .	
	0 ... 1000 V	Nominal supply voltage.	10 = 1 V
48 Brake chopper		Control of the brake chopper.	
48.01	Bc enable	Enables the brake chopper control. Note: Before enabling the brake chopper control, ensure that a brake resistor is connected and the overvoltage control is switched off (parameter 47.01 Overvolt ctrl).	
	Disable	Brake chopper control disabled.	0
	EnableTherm	Brake chopper control enabled with resistor overload protection.	1
	Enable	Brake chopper control enabled without resistor overload protection. This setting can be used, for example, if the resistor is equipped with a thermal circuit breaker that is wired to stop the drive if the resistor overheats.	2

No.	Name/Value	Description	FbEq
48.02	Bc run-time ena	Selects the source for quick run-time brake chopper control. 0 = Brake chopper IGBT pulses are cut off 1 = Normal brake chopper IGBT modulation This parameter can be used to program the chopper control to function only when the drive is operating in the generator mode.	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
48.03	BrThermTimeConst	Defines the thermal time constant of the brake resistor for overload protection.	
	0 ... 10000 s	Brake resistor thermal time constant.	1 = 1 s
48.04	Br power max cnt	Defines the maximum continuous braking power which will raise the resistor temperature to the maximum allowed value. The value is used in the overload protection.	
	0.0000 ... 10000.0000 kW	Maximum continuous braking power.	10000 = 1 kW
48.05	R br	Defines the resistance value of the brake resistor. The value is used for brake chopper protection.	
	0.1000 ... 1000.0000 ohm	Brake resistor resistance value.	10000 = 1 ohm
48.06	Br temp faultlim	Selects the fault limit for the brake resistor temperature supervision. The value is given in percent of the temperature the resistor reaches when loaded with the power defined by parameter 48.04 Br power max cnt . When the limit is exceeded the drive trips on fault BR OVERHEAT.	
	0 ... 150%	Brake resistor temperature fault limit.	1 = 1%
48.07	Br temp alarmlim	Selects the alarm limit for the brake resistor temperature supervision. The value is given in percent of the temperature the resistor reaches when loaded with the power defined by parameter 48.04 Br power max cnt . When the limit is exceeded, the drive generates a BR OVERHEAT alarm.	
	0 ... 150%	Brake resistor temperature alarm limit.	1 = 1%
49 Data storage		16- and 32-bit data storage parameters that can be written to and read from using other parameters' pointer settings. See also section Data storage parameters on page 88.	
49.01	Data storage1	Data storage parameter 1.	
	-32768 ... 32767	16-bit data.	1 = 1
49.02	Data storage2	Data storage parameter 2.	
	-32768 ... 32767	16-bit data.	1 = 1
49.03	Data storage3	Data storage parameter 3.	
	-32768 ... 32767	16-bit data.	1 = 1
49.04	Data storage4	Data storage parameter 4.	
	-32768 ... 32767	16-bit data.	1 = 1
49.05	Data storage5	Data storage parameter 5.	
	-2147483647 ... 2147483647	32-bit data.	1 = 1

No.	Name/Value	Description	FbEq
49.06	Data storage6	Data storage parameter 6.	
	-2147483647 ... 2147483647	32-bit data.	1 = 1
49.07	Data storage7	Data storage parameter 7.	
	-2147483647 ... 2147483647	32-bit data.	1 = 1
49.08	Data storage8	Data storage parameter 8.	
	-2147483647 ... 2147483647	32-bit data.	1 = 1
50 Fieldbus		Settings for configuration of communication via a fieldbus adapter. See also chapter <i>Control through a fieldbus adapter</i> on page 331.	
50.01	Fba enable	Enables communication between the drive and fieldbus adapter.	
	Disable	Communication between the drive and fieldbus adapter disabled.	0
	Enable	Communication between the drive and fieldbus adapter enabled.	1
50.02	Comm loss func	Selects how the drive reacts upon a fieldbus communication break. The time delay is defined by parameter <i>50.03 Comm loss t out</i> .	
	No	Communication break detection disabled.	0
	Fault	Communication break detection active. Upon a communication break, the drive trips on fault FIELDBUS COMM and coasts to stop.	1
	Spd ref Safe	Communication break detection active. Upon a communication break, the drive generates alarm FIELDBUS COMM and sets the speed to the value defined by parameter <i>30.02 Speed ref safe</i> .  WARNING! Make sure that it is safe to continue operation in case of a communication break.	2
	Last speed	Communication break detection active. Upon a communication break, the drive generates alarm FIELDBUS COMM and freezes the speed to the level the drive was operating at. The speed is determined by the average speed over the previous 10 seconds.  WARNING! Make sure that it is safe to continue operation in case of a communication break.	3
50.03	Comm loss t out	Defines the time delay before the action defined by parameter <i>50.02 Comm loss func</i> is taken. Time count starts when the link fails to update the message.	
	0.3 ... 6553.5 s	Time delay.	10 = 1 s
50.04	Fb ref1 modesel	Selects the fieldbus reference FBA REF1 scaling and the actual value, which is sent to the fieldbus (FBA ACT1).	
	Raw data	No scaling (i.e. data is transmitted without scaling). Source for the actual value, which is sent to the fieldbus, is selected by parameter <i>50.06 Fb act1 tr src</i> .	0

No.	Name/Value	Description	FbEq
	Torque	Fieldbus uses torque reference scaling. Torque reference scaling is defined by the used fieldbus profile (e.g. with ABB Drives Profile integer value 10000 corresponds to 100% torque value). Signal 01.06 Motor torque is sent to the fieldbus as an actual value. See the <i>User's Manual</i> of the appropriate fieldbus adapter module.	1
	Speed	Fieldbus uses speed reference scaling. Speed reference scaling is defined by the used fieldbus profile (e.g. with ABB Drives Profile integer value 20000 corresponds to parameter 19.01 Speed scaling value). Signal 01.01 Motor speed rpm is sent to the fieldbus as an actual value. See the <i>User's Manual</i> of the appropriate fieldbus adapter module.	2
50.05	Fb ref2 modesel	Selects the fieldbus reference FBA REF2 scaling. See parameter 50.04 Fb ref1 modesel .	
	Raw data	See parameter 50.04 Fb ref1 modesel .	0
	Torque	See parameter 50.04 Fb ref1 modesel .	1
	Speed	See parameter 50.04 Fb ref1 modesel .	2
50.06	Fb act1 tr src	Selects the source for fieldbus actual value 1 when parameter 50.04 Fb ref1 modesel / 50.05 Fb ref2 modesel is set to Raw data .	
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
50.07	Fb act2 tr src	Selects the source for fieldbus actual value 2 when parameter 50.04 Fb ref1 modesel / 50.05 Fb ref2 modesel is set to Raw data .	
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
50.08	Fb sw b12 src	Selects the source for freely programmable fieldbus status word bit 28 (02.24 FBA main sw bit 28). Note that this functionality may not be supported by the fieldbus communication profile.	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
50.09	Fb sw b13 src	Selects the source for freely programmable fieldbus status word bit 29 (02.24 FBA main sw bit 29). Note that this functionality may not be supported by the fieldbus communication profile.	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
50.10	Fb sw b14 src	Selects the source for freely programmable fieldbus status word bit 30 (02.24 FBA main sw bit 30). Note that this functionality may not be supported by the fieldbus communication profile.	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		
50.11	Fb sw b15 src	Selects the source for freely programmable fieldbus status word bit 31 (02.24 FBA main sw bit 31). Note that this functionality may not be supported by the fieldbus communication profile.	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		

No.	Name/Value	Description	FbEq						
50.15	Fb cw used	Selects the fieldbus Control Word which controls the drive.							
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-						
50.20	Fb main sw func	Selects the rule on the basis of which the drive defines the value for 02.24 FBA main sw bit 1 (Enabled).							
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Name</th> <th>Information</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Run enable func</td> <td> <p>1 = Parameter only: Drive writes value 1 to the bit when the external run enable signal (par. 10.11 Run enable) has value 1.</p> <p>0 = Param AND Fb cw: Drive writes value 1 to the bit when the external run enable signal (par. 10.11 Run enable) is 1 AND 02.22 FBA main cw bit 7 (Run enable) is 1.</p> </td> </tr> </tbody> </table>	Bit	Name	Information	0	Run enable func	<p>1 = Parameter only: Drive writes value 1 to the bit when the external run enable signal (par. 10.11 Run enable) has value 1.</p> <p>0 = Param AND Fb cw: Drive writes value 1 to the bit when the external run enable signal (par. 10.11 Run enable) is 1 AND 02.22 FBA main cw bit 7 (Run enable) is 1.</p>	
Bit	Name	Information							
0	Run enable func	<p>1 = Parameter only: Drive writes value 1 to the bit when the external run enable signal (par. 10.11 Run enable) has value 1.</p> <p>0 = Param AND Fb cw: Drive writes value 1 to the bit when the external run enable signal (par. 10.11 Run enable) is 1 AND 02.22 FBA main cw bit 7 (Run enable) is 1.</p>							

51 FBA settings		Fieldbus adapter-specific settings.	
51.01	FBA type	Displays the type of the connected fieldbus adapter module. 0 = Fieldbus module is not found or is not properly connected, or parameter 50.01 Fba enable is set to <i>Disable</i> , 1 = FPBA-xx PROFIBUS-DP adapter module, 32 = FCAN-xx CANopen adapter module, 37 = FDNA-xx DeviceNet adapter module	
51.02	FBA par2	Parameters 51.02...51.26 are adapter module-specific. For more information, see the <i>User's Manual</i> of the fieldbus adapter module. Note that not all of these parameters are necessarily used.	-
...
51.26	FBA par26	See parameter 51.02 FBA par2 .	-
51.27	FBA par refresh	Validates any changed adapter module configuration parameter settings. After refreshing, the value reverts automatically to <i>Done</i> . Note: This parameter cannot be changed while the drive is running.	
	Done	Refreshing done.	0
	Refresh	Refreshing.	1
51.28	Par table ver	Displays the parameter table revision of the fieldbus adapter module mapping file stored in the memory of the drive. In format xyz, where x = major revision number; y = minor revision number; z = correction number.	
	0x0000 ... 0xFFFF	Parameter table revision.	1 = 1
51.29	Drive type code	Displays the drive type code of the fieldbus adapter module mapping file stored in the memory of the drive.	
	0 ... 65535	Drive type code of fieldbus adapter module mapping file.	1 = 1
51.30	Mapping file ver	Displays the fieldbus adapter module mapping file revision stored in the memory of the drive in decimal format. Example: 0x107 = revision 1.07.	
	0 ... 65535	Mapping file revision.	1 = 1
51.31	D2FBA comm sta	Displays the status of the fieldbus adapter module communication.	
	Idle	Adapter is not configured.	0
	Exec.init	Adapter is initializing.	1

No.	Name/Value	Description	FbEq
	Time out	A timeout has occurred in the communication between the adapter and the drive.	2
	Conf.err	Adapter configuration error: The major or minor revision code of the common program revision in the fieldbus adapter module is not the revision required by the module (see parameter 51.32 FBA comm sw ver) or mapping file upload has failed more than three times.	3
	Off-line	Adapter is off-line.	4
	On-line	Adapter is on-line.	5
	Reset	Adapter is performing a hardware reset.	6
51.32	FBA comm sw ver	Displays the common program revision of the adapter module in format axyz, where a = major revision number, xy = minor revision numbers. z = correction letter. Example: 190A = revision 1.90A.	
		Common program version of adapter module.	1 = 1
51.33	FBA appl sw ver	Displays the application program revision of the adapter module in format axyz, where: a = major revision number, xy = minor revision numbers, z = correction letter. Example: 190A = revision 1.90A.	
		Application program revision of adapter module.	1 = 1
52 FBA data in		Selection of data to be transferred from drive to fieldbus controller via fieldbus adapter.	
52.01	FBA data in1	Parameters 52.01...52.12 select data to be transferred from the drive to the fieldbus controller.	
	4	Status Word (16 bits)	4
	5	Actual value 1 (16 bits)	5
	6	Actual value 2 (16 bits)	6
	14	Status Word (32 bits)	14
	15	Actual value 1 (32 bits)	15
	16	Actual value 2 (32 bits)	16
	101...9999	Parameter index	1 = 1
...
52.12	FBA data in12	See parameter 52.01 FBA data in1 .	

234 Parameters

No.	Name/Value	Description	FbEq
53 FBA data out		Selection of data to be transferred from fieldbus controller to drive via fieldbus adapter.	
53.01	FBA data out1	Parameters 53.01 ... 53.12 select data to be transferred from the fieldbus controller to the drive.	
	1	Control Word (16 bits)	1
	2	Reference REF1 (16 bits)	2
	3	Reference REF2 (16 bits)	3
	11	Control Word (32 bits)	11
	12	Reference REF1 (32 bits)	12
	13	Reference REF2 (32 bits)	13
	101...9999	Parameter index	1 = 1
...
53.12	FBA data out12	See parameter 53.01 FBA data out1 .	
56 Panel display		Selection of signals to be displayed on control panel.	
56.01	Signal1 param	Selects the first signal to be displayed on the optional control panel. The default signal is 01.03 Output frequency .	
	00.00 ... 255.255	1st signal to be displayed.	-
56.02	Signal2 param	Selects the second signal to be displayed on the optional control panel. The default signal is 01.04 Motor current .	
	00.00 ... 255.255	2nd signal to be displayed.	-
56.03	Signal3 param	Selects the third signal to be displayed on the optional control panel. The default signal is 01.06 Motor torque .	
	00.00 ... 255.255	3rd signal to be displayed.	-
56.04	Signal1 mode	Defines the way the signal selected by parameter 56.01 Signal1 param is displayed on the optional control panel.	
	Disabled	Signal not displayed. Any other signals that are not disabled are shown together with their respective signal name.	-1
	Normal	Shows the signal as a numerical value followed by unit.	0
	Bar	Shows the signal as a horizontal bar.	1
	Drive name	Shows the drive name. (The drive name can be set using the DriveStudio PC tool.)	2
	Drive type	Shows the drive type.	3
56.05	Signal2 mode	Defines the way the signal selected by parameter 56.02 Signal2 param is displayed on the optional control panel.	
	Disabled	Signal not displayed. Any other signals that are not disabled are shown together with their respective signal name.	-1
	Normal	Shows the signal as a numerical value followed by unit.	0
	Bar	Shows the signal as a horizontal bar.	1
	Drive name	Shows the drive name. (The drive name can be set using the DriveStudio PC tool.)	2
	Drive type	Shows the drive type.	3

No.	Name/Value	Description	FbEq
56.06	Signal3 mode	Defines the way the signal selected by parameter 56.03 Signal3 param is displayed on the optional control panel.	
	Disabled	Signal not displayed. Any other signals that are not disabled are shown together with their respective signal name.	-1
	Normal	Shows the signal as a numerical value followed by unit.	0
	Bar	Shows the signal as a horizontal bar.	1
	Drive name	Shows the drive name. (The drive name can be set using the DriveStudio PC tool.)	2
	Drive type	Shows the drive type.	3
56.07	Local ref unit	Defines how speed reference is entered and displayed by the optional control panel and DriveStudio PC tool. Also determines the unit of signal 02.34 Panel ref . Note: This parameter also applies to external control when speed reference is given from the control panel.	
	rpm	Speed reference is displayed and entered in rpm.	0
	Percent	Speed reference is displayed and entered in percent. The scaling is as follows: <div style="text-align: center;"> <p>Control panel reference Speed (rpm)</p> <p>100% — 20.01 Maximum speed</p> <p>0% — 0</p> <p>-100% — 20.02 Minimum speed</p> </div>	1
57 D2D communication		Configuration of drive-to-drive communication. See also chapter Drive-to-drive link on page 341 .	
57.01	Link mode	Activates the drive-to-drive connection. Note: Drive-to-drive connection can be enabled only if the embedded fieldbus interface is disabled (parameter 58.01 Protocol ena sel is set to Disabled).	
	Disabled	Drive-to-drive connection disabled.	0
	Follower	The drive is a follower on the drive-to-drive link.	1
	Master	The drive is the master on the drive-to-drive link. Only one drive can be the master at a time.	2
57.02	Comm loss func	Selects how the drive acts when an erroneous drive-to-drive configuration or a communication break is detected.	
	No	Protection not active.	0
	Alarm	The drive generates an alarm.	1
	Fault	The drive trips on a fault.	2
57.03	Node address	Sets the node address for a follower drive. Each follower must have a dedicated node address. Note: If the drive is set to be the master on the drive-to-drive link, this parameter has no effect (the master is automatically assigned node address 0).	
	1 ... 62	Node address.	1 = 1

No.	Name/Value	Description	FbEq
57.04	Follower mask 1	On the master drive, selects the followers to be polled. If no response is received from a polled follower, the action selected by parameter 57.02 Comm loss func is taken. The least significant bit represents follower with node address 1, while the most significant bit represents follower 31. When a bit is set to 1, the corresponding node address is polled. For example, followers 1 and 2 are polled when this parameter is set to the value of 0x3.	
	0h00000000 ... 0h7FFFFFFF	Follower mask 1.	1 = 1
57.05	Follower mask 2	On the master drive, selects the followers to be polled. If no response is received from a polled follower, the action selected by parameter 57.02 Comm loss func is taken. The least significant bit represents follower with node address 32, while the most significant bit represents follower 62. When a bit is set to 1, the corresponding node address is polled. For example, followers 32 and 33 are polled when this parameter is set to the value of 0x3.	
	0h00000000 ... 0h7FFFFFFF	Follower mask 2.	1 = 1
57.06	Ref 1 src	Selects the source of D2D reference 1 sent to the followers. The parameter is effective on the master drive, as well as submasters (57.03 Node address = 57.12 Ref1 mc group) in a multicast message chain (see parameter 57.11 Ref1 msg type)..	
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
57.07	Ref 2 src	On the master drive, selects the source of D2D reference 2 broadcast to all followers.	
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
57.08	Follower cw src	Selects the source of the D2D control word sent to the followers. The parameter is effective on the master drive, as well as submasters in a multicast message chain (see parameter 57.11 Ref1 msg type)..	
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
57.11	Ref1 msg type	By default, in drive-to-drive communication, the master broadcasts the drive-to-drive control word and references 1 and 2 to all followers. This parameter enables multicasting, i.e. sending the drive-to-drive control word and reference 1 to a certain drive or group of drives. The message can then be further relayed to another group of drives to form a multicast chain. In the master, as well as any submaster (i.e. follower relaying the message to other followers), the sources for the control word and reference 1 are selected by parameters 57.08 Follower cw src and 57.06 Ref 1 src respectively. Note: Reference 2 is broadcast by the master to all followers.	
	Broadcast	The control word and reference 1 are sent by the master to all followers. If the master has this setting, the parameter has no effect in the followers.	0

No.	Name/Value	Description	FbEq
	Ref1 MC Grps	The drive-to-drive control word and reference 1 are only sent to the drives in the multicast group specified by parameter 57.13 Next ref1 mc grp . This setting can also be used in submasters (followers in which parameters 57.03 Node address and 57.12 Ref1 mc group are set to the same value) to form a multicast chain.	1
57.12	Ref1 mc group	Selects the multicast group the drive belongs to. See parameter 57.11 Ref1 msg type .	
	0...62	Multicast group.	1 = 1
57.13	Next ref1 mc grp	Specifies the next multicast group of drives the multicast message is relayed to. See parameter 57.11 Ref1 msg type . This parameter is effective only in the master or submasters (followers in which parameters 57.03 Node address and 57.12 Ref1 mc group are set to the same value).	
	0	No group selected.	0
	1...62	Next multicast group in the chain.	1 = 1
57.14	Nr ref1 mc grps	Sets the number of drives sending messages in the message chain. The value is typically equal to the number of multicast groups in the chain assuming that the last drive is NOT sending an acknowledgement to the master. See parameter 57.11 Ref1 msg type . Note: This parameter is only effective in the master.	
	1...62	Number of links in the multicast chain.	1 = 1
57.15	D2D com port	Defines the hardware to which the drive-to-drive link is connected. In special cases (such as harsh operating conditions), the FMBA module may make for more robust communication than the standard drive-to-drive connection.	
	on-board	Connector XD2D on the JCU Control Unit is used.	0
	Slot 1	An FMBA module installed in JCU option slot 1 is used.	1
	Slot 2	An FMBA module installed in JCU option slot 2 is used.	2
	Slot 3	An FMBA module installed in JCU option slot 3 is used.	3
58 Embedded Modbus		Configuration parameters for the embedded fieldbus (EFB) interface. See also chapter Control through the embedded fieldbus interface on page 303.	
58.01	Protocol ena sel	Enables/disables the embedded fieldbus communication protocol. Note: When the embedded fieldbus interface is enabled, the drive-to-drive link operation (parameter group 57) is automatically disabled.	
	Disabled	Disabled.	0
	Modbus RTU	Modbus RTU protocol enabled.	1
58.03	Node address	Defines the node address.	
	0...247	Node address.	1 = 1
58.04	Baud rate	Selects the baud rate of the RS-485 link.	
	4800	4.8 kbit/s.	0
	9600	9.6 kbit/s.	1
	19200	19.2 kbit/s.	2
	38400	38.4 kbit/s.	3

No.	Name/Value	Description	FbEq
	57600	57.6 kbit/s.	4
	76800	76.8 kbit/s.	5
	115200	115.2 kbit/s.	6
58.05	Parity	Selects the number of the data bits, the use and type of the parity bit, and the number of the stop bits.	
	8 none 1	Eight data bits, no parity bit, one stop bit.	0
	8 none 2	Eight data bits, no parity bit, two stop bits.	1
	8 even 1	Eight data bits, even parity bit, one stop bit.	2
	8 odd 1	Eight data bits, odd parity bit, one stop bit.	3
58.06	Control profile	Selects the communication profile used by the Modbus protocol.	
	ABB Classic	ABB Drives profile, classic version.	0
	ABB Enhanced	ABB Drives profile, enhanced version.	1
	DCU 16-bit	DCU 16-bit profile.	2
	DCU 32-bit	DCU 32-bit profile.	3
58.07	Comm loss t out	Defines the timeout limit for EFB communication loss monitoring. If a communication break exceeds the timeout limit, the function proceeds with the action defined with parameter 58.09 Comm loss action . See also parameter 58.08 Comm loss mode .	
	0...60000 ms	Timeout calculation factor. The actual timeout value is calculated as follows: Comm loss timeout × 100 ms Example: If you set this value to 22, the actual timeout value will be: 22 × 100 ms = 2 200 ms.	100 = 1 ms
58.08	Comm loss mode	Enables/disables EFB communication loss monitoring and defines which of the Modbus register accesses resets the timeout counter. See parameter 58.07 Comm loss t out .	
	None	EFB communication loss monitoring is disabled.	0
	Any message	EFB communication loss monitoring is enabled. Any Modbus request resets the timeout counter.	1
	Ctrl write	EFB communication loss monitoring is enabled. Writing to control or reference word resets the timeout counter.	2
58.09	Comm loss action	Defines the drive operation after the EFB communication loss monitoring awakes. See parameters 58.07 Comm loss t out and 58.08 Comm loss mode .	
	None	No action.	0
	Fault	Drive trips on a fault (EFB COMM LOSS FAULT).	1
	Safe speed	Drive generates an alarm (EFB COMM LOSS ALARM) and takes the safe speed into use (see parameter 30.02 Speed ref safe).	2
	Last speed	Drive generates an alarm (EFB COMM LOSS ALARM) and takes the last speed into use (average over the previous 10 seconds).	3
58.10	Refresh settings	Refreshes the settings of parameters 58.01...58.09 .	
	Done	Initial value. The value is restored after the refresh is done.	0
	Refresh	Refresh.	1

No.	Name/Value	Description	FbEq																																				
58.11	Reference scale	Defines the factor which the DCU 16-bit communication profile uses when scaling fieldbus references to drive references and drive actual values to fieldbus actual signals. The references are multiplied by this scaling factor. See section <i>DCU 16-bit profile</i> on page 322.																																					
	1...65535	Scaling factor.	1 = 1																																				
58.15	Comm diagnostics	16-bit packed boolean data word for the communication diagnostics flag bits. Read only.																																					
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Information</th> </tr> </thead> <tbody> <tr><td>0</td><td>Not used.</td></tr> <tr><td>1</td><td>Last received packet was not for this node.</td></tr> <tr><td>2</td><td>Not used.</td></tr> <tr><td>3</td><td>At least one packet has been successfully received after the power up.</td></tr> <tr><td>4</td><td>Not used.</td></tr> <tr><td>5</td><td>Communication time-out has occurred.</td></tr> <tr><td>6</td><td>Not used.</td></tr> <tr><td>7</td><td>Not used.</td></tr> <tr><td>8</td><td>Last write was not successful because of a parameter value limit violation.</td></tr> <tr><td>9</td><td>Last read was not successful because only one register was used to read a 32-bit value.</td></tr> <tr><td>10</td><td>Last write was not successful because the parameter was read only.</td></tr> <tr><td>11</td><td>Last parameter access was not successful because the parameter or group did not exist.</td></tr> <tr><td>12</td><td>Not used.</td></tr> <tr><td>13</td><td>Not used.</td></tr> <tr><td>14</td><td>Not used.</td></tr> <tr><td>15</td><td>Last write was not successful because only one register was used to read a 32-bit value.</td></tr> <tr><td>16...31</td><td>Not used.</td></tr> </tbody> </table>	Bit	Information	0	Not used.	1	Last received packet was not for this node.	2	Not used.	3	At least one packet has been successfully received after the power up.	4	Not used.	5	Communication time-out has occurred.	6	Not used.	7	Not used.	8	Last write was not successful because of a parameter value limit violation.	9	Last read was not successful because only one register was used to read a 32-bit value.	10	Last write was not successful because the parameter was read only.	11	Last parameter access was not successful because the parameter or group did not exist.	12	Not used.	13	Not used.	14	Not used.	15	Last write was not successful because only one register was used to read a 32-bit value.	16...31	Not used.	
Bit	Information																																						
0	Not used.																																						
1	Last received packet was not for this node.																																						
2	Not used.																																						
3	At least one packet has been successfully received after the power up.																																						
4	Not used.																																						
5	Communication time-out has occurred.																																						
6	Not used.																																						
7	Not used.																																						
8	Last write was not successful because of a parameter value limit violation.																																						
9	Last read was not successful because only one register was used to read a 32-bit value.																																						
10	Last write was not successful because the parameter was read only.																																						
11	Last parameter access was not successful because the parameter or group did not exist.																																						
12	Not used.																																						
13	Not used.																																						
14	Not used.																																						
15	Last write was not successful because only one register was used to read a 32-bit value.																																						
16...31	Not used.																																						
	0x0000...0xFFFF	Data word (hex).	1 = 1																																				
58.16	Received packets	Shows the number of message packets received by the drive, including only such packets that are addressed to the drive. Note: The user can reset the counter (by setting the value to 0).																																					
	0...65535	No. of message packets.	1 = 1																																				
58.17	Transm packets	Shows the number of message packets sent by the drive. Note: The user can reset the counter (by setting the value to 0).																																					
	0...65535	No. of message packets.	1 = 1																																				
58.18	All packets	Shows the total number of message packets received by the drive, including all packets addressed to any valid node on the fieldbus link. Note: The user can reset the counter (by setting the value to 0).																																					
	0...65535	No. of message packets.	1 = 1																																				
58.19	UART errors	Shows the number of messages with communication errors other than CRC errors which the drive has received (e.g. UART buffer overflow errors). Read only.																																					
	0..65535	No. of messages with errors (excluding messages with CRC errors).	1 = 1																																				

240 Parameters

No.	Name/Value	Description	FbEq
58.20	CRC errors	Shows the number of messages with Cyclic Redundancy Check (CRC) errors which the drive has received. Read only. Note: High electromagnetic noise levels may generate errors.	
	0...65535	No. of messages with CRC errors.	1 = 1
58.21	Raw CW LSW	Shows the LSW part of the Control Word which the drive receives from the Modbus master. Read only.	
	0x0000...0xFFFF	Bits 0...15 of the Control word as a hex value.	1 = 1
58.22	Raw CW MSW	Shows the MSW part of the Control Word which the drive receives from the Modbus master. Read only.	
	0x0000...0xFFFF	Bits 16...32 of the Control word as a hex value.	1 = 1
58.23	Raw SW LSW	Shows the LSW part of the Status Word which the drive sends to the Modbus master. Read only.	
	0x0000...0xFFFF	Bits 0...15 of the Status word as a hex value.	1 = 1
58.24	Raw SW MSW	Shows the MSW part of the Status Word which the drive sends to the Modbus master. Read only.	
	0x0000...0xFFFF	Bits 16...32 of the Status word as a hex value.	1 = 1
58.25	Raw Ref 1 LSW	Shows the LSW part of reference 1 which the drive receives from the Modbus master. Read only.	
	0x0000...0xFFFF	Bits 0...15 of reference 1 as a hex value.	1 = 1
58.26	Raw Ref 1 MSW	Shows the MSW part of reference 1 which the drive receives from the Modbus master. Read only.	
	0x0000...0xFFFF	Bits 16...32 of reference 1 as a hex value.	1 = 1
58.27	Raw Ref 2 LSW	Shows the LSW part of reference 2 which the drive receives from the Modbus master. Read only.	
	0x0000...0xFFFF	Bits 0...15 of reference 2 as a hex value.	1 = 1
58.28	Raw Ref 2 MSW	Shows the MSW part of reference 2 which the drive receives from the Modbus master. Read only.	
	0x0000...0xFFFF	Bits 16...32 of reference 2 as a hex value.	1 = 1
58.30	Transmit delay	Defines the delay time which the slave waits until it sends a response.	
	0...65335 ms	Transmit delay time.	1 = 1 ms
58.31	Ret app errors	Selects whether the drive returns Modbus exception codes or not.	
	No	No	0
	Yes	Yes	1
58.32	Word order	Defines the order of the data words in the Modbus frame.	
	MSW LSW	Most significant word first, then Least significant word.	0
	LSW MSW	Least significant word first, then Most significant word.	1
58.35	Data I/O 1	Defines the address of the drive parameter which the Modbus master accesses when it reads from or writes to the register address corresponding to Modbus In/Out parameter no. 1. The Modbus master defines the type of the data (input or output). The value is conveyed in a Modbus frame using two 16-bit words. If the drive parameter is a 16-bit value, the LSW (Least significant word) conveys the value. If the drive parameter is a 32-bit value, the next Modbus In/Out parameter is also reserved.	

No.	Name/Value	Description	FbEq
	0...9999	Parameter address. Format: xxyy, where: xx = parameter group yy = parameter index	1 = 1
58.36	Data I/O 2	See parameter 58.35 .	
	0...9999	See parameter 58.35 .	1 = 1
...
58.58	Data I/O 24	See parameter 58.35 .	
	0...9999	See parameter 58.35 .	1 = 1
64 Load analyzer		Peak value and amplitude logger settings. See also section Load analyzer on page 85 .	
64.01	PVL signal	Selects the signal to be monitored by the peak value logger. The signal is filtered using the filtering time specified by parameter 64.02 PVL filt time . The peak value is stored, along with other pre-selected signals at the time, into parameters 64.06...64.11 . Parameter 64.03 Reset loggers resets both the peak value logger and amplitude logger 2. The latest time the loggers were reset is stored into parameter 64.13 .	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
64.02	PVL filt time	Peak value logger filtering time. See parameter 64.01 PVL signal .	
	0.00 ... 120.00 s	Peak value logger filtering time.	100 = 1 s
64.03	Reset loggers	Selects the signal to reset the peak value logger and amplitude logger 2. (Amplitude logger 1 cannot be reset.)	
	Const	Bit pointer setting (see Terms and abbreviations on page 102).	-
	Pointer		

242 Parameters

No.	Name/Value	Description	FbEq
64.04	AL signal	<p>Selects the signal to be monitored by amplitude logger 2. The signal is sampled at 200 ms intervals when the drive is running.</p> <p>The results are displayed by parameters 64.24...64.33. Each parameter represents an amplitude range, and shows what portion of the samples fall within that range.</p> <p>The signal value corresponding to 100% is defined by parameter 64.05 AL signal base.</p> <p>Parameter 64.03 Reset loggers resets both the peak value logger and amplitude logger 2. The latest time the loggers were reset is stored into parameter 64.13.</p> <p>Note: Amplitude logger 1 is fixed to monitor motor current (01.04 Motor current). The results are displayed by parameters 64.14...64.23. 100% of the signal value corresponds to the maximum output current of the drive (see the appropriate <i>Hardware Manual</i>).</p>	
	Speed rpm	01.01 Motor speed rpm (see page 103).	1073742081
	Speed %	01.02 Motor speed % (see page 103).	1073742082
	Frequency	01.03 Output frequency (see page 103).	1073742083
	Current	01.04 Motor current (see page 103).	1073742084
	Current %	01.05 Motor current % (see page 103).	1073742085
	Torque	01.06 Motor torque (see page 103).	1073742086
	Dc-voltage	01.07 Dc-voltage (see page 103).	1073742087
	Power inu	01.22 Power inu out (see page 103).	1073742102
	Power motor	01.23 Motor power (see page 103).	1073742103
	Process act	04.03 Process act (see page 115).	1073742851
	Proc PID out	04.05 Process PID out (see page 115).	1073742853
	Pointer	Value pointer setting (see Terms and abbreviations on page 102).	-
64.05	AL signal base	Defines the signal value that corresponds to 100% amplitude.	
	0.00 ... 32768.00	Signal value corresponding to 100%.	100 = 1
64.06	PVL peak value1	Peak value recorded by the peak value logger.	
	-32768.00 ... 32768.00	Peak value.	100 = 1
64.07	Date of peak	The date on which the peak value was recorded.	
	01.01.80 ...	Peak occurrence date (dd.mm.yy).	1 = 1 d
64.08	Time of peak	The time at which the peak value was recorded.	
	00:00:00 ... 23:59:59	Peak occurrence time.	1 = 1 s
64.09	Current at peak	Motor current at the moment the peak value was recorded.	
	-32768.00 ... 32768.00 A	Motor current at peak.	100 = 1 A
64.10	Dc volt at peak	Voltage in the intermediate DC circuit of the drive at the moment the peak value was recorded.	
	0.00 ... 2000.00 V	DC voltage at peak.	100 = 1 V
64.11	Speed at peak	Motor speed at the moment the peak value was recorded.	
	-32768.00 ... 32768.00 rpm	Motor speed at peak.	100 = 1 rpm

No.	Name/Value	Description	FbEq
64.12	Date of reset	The date the peak value logger and amplitude logger 2 were last reset.	
	01.01.80 ...	Last reset date of loggers (dd.mm.yy).	1 = 1 d
64.13	Time of reset	The time the peak value logger and amplitude logger 2 were last reset.	
	00:00:00 ... 23:59:59	Last reset time of loggers.	1 = 1 s
64.14	AL1 0 to 10%	Percentage of samples recorded by amplitude logger 1 that fall between 0 and 10%.	
	0.00 ... 100.00%	Amplitude logger 1 samples between 0 and 10%.	100 = 1%
64.15	AL1 10 to 20%	Percentage of samples recorded by amplitude logger 1 that fall between 10 and 20%.	
	0.00 ... 100.00%	Amplitude logger 1 samples between 10 and 20%.	100 = 1%
64.16	AL1 20 to 30%	Percentage of samples recorded by amplitude logger 1 that fall between 20 and 30%.	
	0.00 ... 100.00%	Amplitude logger 1 samples between 20 and 30%.	100 = 1%
64.17	AL1 30 to 40%	Percentage of samples recorded by amplitude logger 1 that fall between 30 and 40%.	
	0.00 ... 100.00%	Amplitude logger 1 samples between 30 and 40%.	100 = 1%
64.18	AL1 40 to 50%	Percentage of samples recorded by amplitude logger 1 that fall between 40 and 50%.	
	0.00 ... 100.00%	Amplitude logger 1 samples between 40 and 50%.	100 = 1%
64.19	AL1 50 to 60%	Percentage of samples recorded by amplitude logger 1 that fall between 50 and 60%.	
	0.00 ... 100.00%	Amplitude logger 1 samples between 50 and 60%.	100 = 1%
64.20	AL1 60 to 70%	Percentage of samples recorded by amplitude logger 1 that fall between 60 and 70%.	
	0.00 ... 100.00%	Amplitude logger 1 samples between 60 and 70%.	100 = 1%
64.21	AL1 70 to 80%	Percentage of samples recorded by amplitude logger 1 that fall between 70 and 80%.	
	0.00 ... 100.00%	Amplitude logger 1 samples between 70 and 80%.	100 = 1%
64.22	AL1 80 to 90%	Percentage of samples recorded by amplitude logger 1 that fall between 80 and 90%.	
	0.00 ... 100.00%	Amplitude logger 1 samples between 80 and 90%.	100 = 1%
64.23	AL1 over 90%	Percentage of samples recorded by amplitude logger 1 that exceed 90%.	
	0.00 ... 100.00%	Amplitude logger 1 samples over 90%.	100 = 1%
64.24	AL2 0 to 10%	Percentage of samples recorded by amplitude logger 2 that fall between 0 and 10%.	
	0.00 ... 100.00%	Amplitude logger 2 samples between 0 and 10%.	100 = 1%
64.25	AL2 10 to 20%	Percentage of samples recorded by amplitude logger 2 that fall between 10 and 20%.	
	0.00 ... 100.00%	Amplitude logger 2 samples between 10 and 20%.	100 = 1%
64.26	AL2 20 to 30%	Percentage of samples recorded by amplitude logger 2 that fall between 20 and 30%.	
	0.00 ... 100.00%	Amplitude logger 2 samples between 20 and 30%.	100 = 1%

244 Parameters

No.	Name/Value	Description	FbEq
64.27	AL2 30 to 40%	Percentage of samples recorded by amplitude logger 2 that fall between 30 and 40%.	
	0.00 ... 100.00%	Amplitude logger 2 samples between 30 and 40%.	100 = 1%
64.28	AL2 40 to 50%	Percentage of samples recorded by amplitude logger 2 that fall between 40 and 50%.	
	0.00 ... 100.00%	Amplitude logger 2 samples between 40 and 50%.	100 = 1%
64.29	AL2 50 to 60%	Percentage of samples recorded by amplitude logger 2 that fall between 50 and 60%.	
	0.00 ... 100.00%	Amplitude logger 2 samples between 50 and 60%.	100 = 1%
64.30	AL2 60 to 70%	Percentage of samples recorded by amplitude logger 2 that fall between 60 and 70%.	
	0.00 ... 100.00%	Amplitude logger 2 samples between 60 and 70%.	100 = 1%
64.31	AL2 70 to 80%	Percentage of samples recorded by amplitude logger 2 that fall between 70 and 80%.	
	0.00 ... 100.00%	Amplitude logger 2 samples between 70 and 80%.	100 = 1%
64.32	AL2 80 to 90%	Percentage of samples recorded by amplitude logger 2 that fall between 80 and 90%.	
	0.00 ... 100.00%	Amplitude logger 2 samples between 80 and 90%.	100 = 1%
64.33	AL2 over 90%	Percentage of samples recorded by amplitude logger 2 that exceed 90%.	
	0.00 ... 100.00%	Amplitude logger 2 samples over 90%.	100 = 1%
90 Enc module sel		Activation of encoder/resolver interfaces. See also section Encoder support on page 63.	
90.01	Encoder 1 sel	Activates the communication to optional encoder/resolver interface 1. Note: It is recommended that encoder interface 1 is used whenever possible since the data received through that interface is fresher than the data received through interface 2. On the other hand, when position values used in emulation are determined by the drive software, the use of encoder interface 2 is recommended as the values are transmitted earlier through interface 2 than through interface 1.	
	None	Inactive.	0
	FEN-01 TTL+	Communication active. Module type: FEN-01 TTL Encoder interface. Input: TTL encoder input with commutation support (X32).	1
	FEN-01 TTL	Communication active. Module type: FEN-01 TTL Encoder Interface. Input: TTL encoder input (X31).	2
	FEN-11 ABS	Communication active. Module type: FEN-11 Absolute Encoder Interface. Input: Absolute encoder input (X42).	3
	FEN-11 TTL	Communication active. Module type: FEN-11 Absolute Encoder Interface. Input: TTL encoder input (X41).	4
	FEN-21 RES	Communication active. Module type: FEN-21 Resolver Interface. Input: Resolver input (X52).	5
	FEN-21 TTL	Communication active. Module type: FEN-21 Resolver Interface. Input: TTL encoder input (X51).	6
	FEN-31 HTL	Communication active. Module type: FEN-31 HTL Encoder Interface. Input: HTL encoder input (X82).	7

No.	Name/Value	Description	FbEq
90.02	Encoder 2 sel	Activates the communication to the optional encoder/resolver interface 2. Note: The counting of shaft revolutions is not supported for encoder 2.	
	None	Inactive.	0
	FEN-01 TTL+	See parameter 90.01 Encoder 1 sel.	1
	FEN-01 TTL	See parameter 90.01 Encoder 1 sel.	2
	FEN-11 ABS	See parameter 90.01 Encoder 1 sel.	3
	FEN-11 TTL	See parameter 90.01 Encoder 1 sel.	4
	FEN-21 RES	See parameter 90.01 Encoder 1 sel.	5
	FEN-21 TTL	See parameter 90.01 Encoder 1 sel.	6
	FEN-31 HTL	See parameter 90.01 Encoder 1 sel.	7
90.04	TTL echo sel	Enables and selects the interface for the TTL encoder signal echo. Note: If encoder emulation and echo are enabled for the same FEN-xx TTL output, the emulation overrides the echo.	
	Disabled	No echo interface enabled.	0
	FEN-01 TTL+	Module type: FEN-01 TTL Encoder Interface. Echo: TTL encoder input (X32) pulses are echoed to the TTL output.	1
	FEN-01 TTL	Module type: FEN-01 TTL Encoder Interface. Echo: TTL encoder input (X31) pulses are echoed to the TTL output.	2
	FEN-11 TTL	Module type: FEN-11 Absolute Encoder Interface. Echo: TTL encoder input (X41) pulses are echoed to the TTL output.	3
	FEN-21 TTL	Module type: FEN-21 Resolver Interface. Echo: TTL encoder input (X51) pulses are echoed to the TTL output.	4
	FEN-31 HTL	Module type: FEN-31 HTL Encoder Interface. Echo: HTL encoder input (X82) pulses are echoed to the TTL output.	5
90.05	Enc cable fault	Selects the action in case an encoder cable fault is detected by the FEN-xx encoder interface.	
	No	Cable fault detection inactive.	0
	Fault	The drive trips on an ENCODER 1/2 CABLE fault.	1
	Warning	The drive generates an ENCODER 1/2 CABLE warning. This is the recommended setting if the maximum pulse frequency of sine/cosine incremental signals exceeds 100 kHz; at high frequencies, the signals may attenuate enough to invoke the function. The maximum pulse frequency can be calculated as follows: $\text{Max. pulse frequency} = \frac{\text{Pulses per rev.} \times \text{Max. speed in rpm}}{60}$	2
90.10	Enc par refresh	Setting this parameter to 1 forces a reconfiguration of the FEN-xx interfaces, which is needed for any parameter changes in groups 90...93 to take effect. Note: The parameter cannot be changed while the drive is running.	
	Done	Reconfiguration done.	0
	Configure	Reconfigure. The value will automatically revert to <i>Done</i> .	1

No.	Name/Value	Description	FbEq
91	Absol enc conf	Absolute encoder configuration. See also section Encoder support on page 63.	
91.01	Sine cosine nr	Defines the number of sine/cosine wave cycles within one revolution. Note: This parameter does not need to be set when EnDat or SSI encoders are used in continuous mode. See parameter 91.25 SSI mode / 91.30 Endat mode .	
	0...65535	Number of sine/cosine wave cycles.	1 = 1
91.02	Abs enc interf	Selects the source for the encoder absolute position.	
	None	Not selected.	0
	Commut sig	Commutation signals.	1
	EnDat	Serial interface: EnDat encoder.	2
	Hiperface	Serial interface: HIPERFACE encoder.	3
	SSI	Serial interface: SSI encoder.	4
	Tamag. 17/33b	Serial interface: Tamagawa 17/33-bit encoder.	5
91.03	Rev count bits	Defines the number of bits used in revolution count with multiturn encoders. Used when parameter 91.02 Abs enc interf is set to EnDat , Hiperface or SSI . When parameter 91.02 Abs enc interf is set to Tamag. 17/33b , setting this parameter to a non-zero value activates multiturn data requesting.	
	0...32	Number of bits used in revolution count. For example, 4096 revolutions corresponds to 12 bits.	1 = 1
91.04	Pos data bits	Defines the number of bits used within one revolution when parameter 91.02 Abs enc interf is set to EnDat , Hiperface or SSI . When parameter 91.02 Abs enc interf is set to Tamag. 17/33b , this parameter is internally set to 17.	
	0...32	Number of bits. For example, 32768 positions per revolution corresponds to 15 bits.	1 = 1
91.05	Refmark ena	Enables the encoder zero pulse for the absolute encoder input (X42) of an FEN-11 module (if present). Zero pulse can be used for position latching. Note: With serial interfaces (i.e. when parameter 91.02 Abs enc interf setting is EnDat , Hiperface , SSI or Tamag. 17/33b), zero pulse must be disabled.	
	False	Zero pulse disabled.	0
	True	Zero pulse enabled.	1
91.10	Hiperface parity	Defines the use of parity and stop bits for HIPERFACE encoder (i.e. when parameter 91.02 Abs enc interf setting is Hiperface). Typically, this parameter does not need to be set.	
	Odd	Odd parity indication bit, one stop bit.	0
	Even	Even parity indication bit, one stop bit.	1
91.11	Hiperf baudrate	Defines the transfer rate of the link for HIPERFACE encoder (i.e. when parameter 91.02 Abs enc interf setting is Hiperface). Typically, this parameter does not need to be set.	
	4800	4800 bit/s	0
	9600	9600 bit/s	1
	19200	19200 bit/s	2

No.	Name/Value	Description	FbEq
	38400	38400 bit/s	3
91.12	Hiperf node addr	Defines the node address for HIPERFACE encoder (i.e. when parameter <i>91.02 Abs enc interf</i> setting is <i>Hiperface</i>). Typically, this parameter does not need to be set.	
	0...255	HIPERFACE encoder node address.	1 = 1
91.20	SSI clock cycles	Defines the length of the SSI message. The length is defined as the number of clock cycles. The number of cycles can be calculated by adding 1 to the number of bits in an SSI message frame. Used with SSI encoders, i.e. when parameter <i>91.02 Abs enc interf</i> setting is <i>SSI</i> .	
	2...127	Length of SSI message.	1 = 1
91.21	SSI position msb	Defines the location of the MSB (most significant bit) of the position data within an SSI message. Used with SSI encoders, i.e. when parameter <i>91.02 Abs enc interf</i> setting is <i>SSI</i> .	
	1...126	Location of MSB (bit number) in SSI position data.	1 = 1
91.22	SSI revol msb	Defines the location of the MSB (most significant bit) of the revolution count within an SSI message. Used with SSI encoders, i.e. when parameter <i>91.02 Abs enc interf</i> setting is <i>SSI</i> .	
	1...126	Location of MSB (bit number) in SSI revolution count.	1 = 1
91.23	SSI data format	Selects the data format for SSI encoder (i.e. when parameter <i>91.02 Abs enc interf</i> setting is <i>SSI</i>).	
	binary	Binary data format.	0
	gray	Gray data format.	1
91.24	SSI baud rate	Selects the baud rate for SSI encoder (i.e. when parameter <i>91.02 Abs enc interf</i> setting is <i>SSI</i>).	
	10 kbit/s	10 kbit/s baud rate.	0
	50 kbit/s	50 kbit/s baud rate.	1
	100 kbit/s	100 kbit/s baud rate.	2
	200 kbit/s	200 kbit/s baud rate.	3
	500 kbit/s	500 kbit/s baud rate.	4
	1000 kbit/s	1000 kbit/s baud rate.	5
91.25	SSI mode	Selects the SSI encoder mode. Note: This parameter needs to be set only when an SSI encoder is used in continuous mode, i.e. without incremental sin/cos signals (supported only as encoder 1). SSI encoder is selected by setting parameter <i>91.02 Abs enc interf</i> to <i>SSI</i> .	
	Initial pos.	Single position transfer mode (initial position).	0
	Continuous	Continuous position transfer mode.	1
91.26	SSI transmit cyc	Selects the transmission cycle for SSI encoder. Note: This parameter needs to be set only when an SSI encoder is used in continuous mode, i.e. without incremental sin/cos signals (supported only as encoder 1). SSI encoder is selected by setting parameter <i>91.02 Abs enc interf</i> to <i>SSI</i> .	
	50 µs	50 µs transmission cycle.	0
	100 µs	100 µs transmission cycle.	1

No.	Name/Value	Description	FbEq
	200 μ s	200 μ s transmission cycle.	2
	500 μ s	500 μ s transmission cycle.	3
	1 ms	1 ms transmission cycle.	4
	2 ms	2 ms transmission cycle.	5
91.27	SSI zero phase	Defines the phase angle within one sine/cosine signal period that corresponds to the value of zero on the SSI serial link data. The parameter is used to adjust the synchronization of the SSI position data and the position based on sine/cosine incremental signals. Incorrect synchronization may cause an error of ± 1 incremental period. Note: This parameter needs only be set when an SSI encoder with sine/cosine incremental signals is used in initial position mode.	
	315-45 deg	315...45° phase angle.	0
	45-135 deg	45...135° phase angle.	1
	135-225 deg	135...225° phase angle.	2
	225-315 deg	225...315° phase angle.	3
91.30	Endat mode	Selects the EnDat encoder mode. Note: This parameter needs to be set only when an EnDat encoder is used in continuous mode, i.e. without incremental sin/cos signals (supported only as encoder 1). EnDat encoder is selected by setting parameter 91.02 Abs enc interf to <i>EnDat</i> .	
	Initial pos.	Single position data transfer (initial position).	0
	Continuous	Continuous position data transfer mode.	1
91.31	Endat max calc	Selects the maximum encoder calculation time for EnDat encoder. Note: This parameter needs to be set only when an EnDat encoder is used in continuous mode, i.e. without incremental sin/cos signals (supported only as encoder 1). EnDat encoder is selected by setting parameter 91.02 Abs enc interf to <i>EnDat</i> .	
	10 μ s	10 μ s maximum calculation time.	0
	100 μ s	100 μ s maximum calculation time.	1
	1 ms	1 ms maximum calculation time.	2
	50 ms	50 ms maximum calculation time.	3
92 Resolver conf		Resolver configuration. See also section Encoder support on page 63 .	
92.01	Resolv polepairs	Selects the number of pole pairs.	
	1 ... 32	Number of pole pairs.	1 = 1
92.02	Exc signal ampl	Defines the amplitude of the excitation signal.	
	4.0 ... 12.0 Vrms	Amplitude of excitation signal.	10 = 1 Vrms
92.03	Exc signal freq	Defines the frequency of the excitation signal.	
	1 ... 20 kHz	Frequency of excitation signal.	1 = 1 kHz

No.	Name/Value	Description	FbEq								
93	Pulse enc conf	Pulse encoder configuration. See also section Encoder support on page 63.									
93.01	Enc1 pulse nr	Defines the pulse number per revolution for encoder 1.									
	0 ... 65535	Number of pulses for encoder 1.	1 = 1								
93.02	Enc1 type	Selects the type of the encoder 1.									
	Quadrature	Quadrature encoder (has two channels, A and B)	0								
	Single track	Single track encoder (has one channel, A)	1								
93.03	Enc1 sp CalcMode	Selects the speed calculation mode for encoder 1.									
	A&B all	Channels A and B: Rising and falling edges are used for speed calculation. Channel B: Defines the direction of rotation. Notes: <ul style="list-style-type: none"> When single track mode has been selected by parameter 93.02 Enc1 type, this setting acts like the setting A all. When single track mode has been selected by parameter 93.02 Enc1 type, the speed is always positive. 	0								
	A all	Channel A: Rising and falling edges are used for speed calculation. Channel B: Defines the direction of rotation. Note: When single track mode has been selected by parameter 93.02 Enc1 type , the speed is always positive.	1								
	A rising	Channel A: Rising edges are used for speed calculation. Channel B: Defines the direction of rotation. Note: When single track mode has been selected by parameter 93.02 Enc1 type , the speed is always positive.	2								
	A falling	Channel A: Falling edges are used for speed calculation. Channel B: Defines the direction of rotation. Note: When single track mode has been selected by parameter 93.02 Enc1 type , the speed is always positive.	3								
	Auto rising	One of the above modes is selected automatically depending on the pulse frequency as follows: <table border="1" data-bbox="545 1375 1260 1518"> <thead> <tr> <th>Pulse frequency of the channel(s)</th> <th>Mode used</th> </tr> </thead> <tbody> <tr> <td>< 2442 Hz</td> <td>A&B all</td> </tr> <tr> <td>2442...4884 Hz</td> <td>A all</td> </tr> <tr> <td>> 4884 Hz</td> <td>A rising</td> </tr> </tbody> </table>	Pulse frequency of the channel(s)	Mode used	< 2442 Hz	A&B all	2442...4884 Hz	A all	> 4884 Hz	A rising	4
Pulse frequency of the channel(s)	Mode used										
< 2442 Hz	A&B all										
2442...4884 Hz	A all										
> 4884 Hz	A rising										
	Auto falling	One of the above modes is selected automatically depending on the pulse frequency as follows: <table border="1" data-bbox="545 1641 1260 1785"> <thead> <tr> <th>Pulse frequency of the channel(s)</th> <th>Mode used</th> </tr> </thead> <tbody> <tr> <td>< 2442 Hz</td> <td>A&B all</td> </tr> <tr> <td>2442...4884 Hz</td> <td>A all</td> </tr> <tr> <td>> 4884 Hz</td> <td>A falling</td> </tr> </tbody> </table>	Pulse frequency of the channel(s)	Mode used	< 2442 Hz	A&B all	2442...4884 Hz	A all	> 4884 Hz	A falling	5
Pulse frequency of the channel(s)	Mode used										
< 2442 Hz	A&B all										
2442...4884 Hz	A all										
> 4884 Hz	A falling										
93.11	Enc2 pulse nr	Defines the pulse number per revolution for encoder 2.									
	0 ... 65535	Number of pulses for encoder 2.	1 = 1								
93.12	Enc2 type	Selects the type of the encoder 2.									
	Quadrature	Quadrature encoder (has two channels, A and B)	0								
	Single track	Single track encoder (has one channel, A)	1								

250 Parameters

No.	Name/Value	Description	FbEq
93.13	Enc2 sp CalcMode	Selects the speed calculation mode for encoder 2.	
	A&B all	See parameter 93.03 Enc1 sp CalcMode .	0
	A all	See parameter 93.03 Enc1 sp CalcMode .	1
	A rising	See parameter 93.03 Enc1 sp CalcMode .	2
	A falling	See parameter 93.03 Enc1 sp CalcMode .	3
	Auto rising	See parameter 93.03 Enc1 sp CalcMode .	4
	Auto falling	See parameter 93.03 Enc1 sp CalcMode .	5
94 Ext IO conf		I/O extension configuration.	
94.01	Ext IO1 sel	Activates an I/O extension module installed into Slot 1. Depending on the module used, enables DI8...DI9, DIO3...DIO10, AI3...AI5, AO3...AO4 or RO4...RO7.	
	None	No extension installed into Slot 1.	0
	FIO-01	FIO-01 extension installed into Slot 1. Extra 4 x DIO and 2 x RO are in use.	1
	FIO-11	FIO-11 extension installed into Slot 1. Extra 3 x AI, 1 x AO and 2 x DIO are in use.	2
	FIO-21	FIO-21 extension installed into Slot 1. Extra 1 x AI, 1 x DI and 2 x RO are in use.	3
94.02	Ext IO2 sel	Activates an I/O extension module installed into Slot 2. Depending on the module used, enables DI8...DI9, DIO3...DIO10, AI3...AI5, AO3...AO4 or RO4...RO7.	
	None	No 2nd extension installed into Slot 2.	0
	FIO-01	FIO-01 extension installed into Slot 2. Extra 4 x DIO and 2 x RO are in use.	1
	FIO-11	FIO-11 extension installed into Slot 2. Extra 3 x AI, 1 x AO and 2 x DIO are in use.	2
	FIO-21	FIO-21 extension installed into Slot 2. Extra 1 x AI, 1 x DI and 2 x RO are in use.	3
95 Hw configuration		Diverse hardware-related settings.	
95.01	Ctrl boardSupply	Selects how the drive control unit is powered.	
	Internal 24V	The drive control unit is powered from the drive power unit it is mounted on. This is the default setting.	0
	External 24V	The drive control unit is powered from an external power supply.	1
95.03	Temp inu ambient	Defines the maximum ambient temperature. This temperature is used to calculate the estimated drive temperature. If the measured drive temperature exceeds this estimated value, an alarm (COOLALARM) or fault (COOLING) is generated.	
	0 ... 55 °C	Drive ambient temperature.	1 = 1 °C


No.	Name/Value	Description	FbEq
97 User motor par		Motor values supplied by the user that are used in the motor model.	
97.01	Use given params	Activates the motor model parameters 97.02...97.14 and the rotor angle offset parameter 97.20. Notes: <ul style="list-style-type: none"> Parameter value is automatically set to zero when ID run is selected by parameter 99.13 <i>IDrun mode</i>. The values of parameters 97.02...97.20 are updated according to the motor characteristics identified during the ID run. This parameter cannot be changed while the drive is running. 	
	NoUserPars	Parameters 97.02...97.20 inactive.	0
	UserMotPars	The values of parameters 97.02...97.14 are used in the motor model.	1
	UserPosOffs	The value of parameter 97.20 is used as the rotor angle offset. Parameters 97.02...97.14 are inactive.	2
	AllUserPars	The values of parameters 97.02...97.14 are used in the motor model, and the value of parameter 97.20 is used as the rotor angle offset.	3
97.02	Rs user	Defines the stator resistance R_S of the motor model.	
	0.00000 ... 0.50000 p.u.	Stator resistance in per unit.	100000 = 1 p.u.
97.03	Rr user	Defines the rotor resistance R_R of the motor model. Note: This parameter is valid only for asynchronous motors.	
	0.00000 ... 0.50000 p.u.	Rotor resistance in per unit.	100000 = 1 p.u.
97.04	Lm user	Defines the main inductance L_M of the motor model. Note: This parameter is valid only for asynchronous motors.	
	0.00000 ... 10.00000 p.u.	Main inductance in per unit.	100000 = 1 p.u.
97.05	SigmaL user	Defines the leakage inductance σL_S . Note: This parameter is valid only for asynchronous motors.	
	0.00000 ... 1.00000 p.u.	Leakage inductance in per unit.	100000 = 1 p.u.
97.06	Ld user	Defines the direct axis (synchronous) inductance. Note: This parameter is valid only for permanent magnet motors.	
	0.00000 ... 10.00000 p.u.	Direct axis inductance in per unit.	100000 = 1 p.u.
97.07	Lq user	Defines the quadrature axis (synchronous) inductance. Note: This parameter is valid only for permanent magnet motors.	
	0.00000 ... 10.00000 p.u.	Quadrature axis inductance in per unit.	100000 = 1 p.u.
97.08	Pm flux user	Defines the permanent magnet flux. Note: This parameter is valid only for permanent magnet motors.	
	0.00000 ... 2.00000 p.u.	Permanent magnet flux in per unit.	100000 = 1 p.u.


No.	Name/Value	Description	FbEq
97.09	Rs user SI	Defines the stator resistance R_S of the motor model.	
	0.00000 ... 100.00000 ohm	Stator resistance.	100000 = 1 ohm
97.10	Rr user SI	Defines the rotor resistance R_R of the motor model. Note: This parameter is valid only for asynchronous motors.	
	0.00000 ... 100.00000 ohm	Rotor resistance.	100000 = 1 ohm
97.11	Lm user SI	Defines the main inductance L_M of the motor model. Note: This parameter is valid only for asynchronous motors.	
	0.00 ...100000.00 mH	Main inductance.	100 = 1 mH
97.12	SigL user SI	Defines the leakage inductance σL_S . Note: This parameter is valid only for asynchronous motors.	
	0.00 ...100000.00 mH	Leakage inductance.	100 = 1 mH
97.13	Ld user SI	Defines the direct axis (synchronous) inductance. Note: This parameter is valid only for permanent magnet motors.	
	0.00 ...100000.00 mH	Direct axis inductance.	100 = 1 mH
97.14	Lq user SI	Defines the quadrature axis (synchronous) inductance. Note: This parameter is valid only for permanent magnet motors.	
	0.00 ...100000.00 mH	Quadrature axis inductance.	100 = 1 mH
97.20	PM angle offset	Defines an angle offset between the zero position of the synchronous motor and the zero position of the position sensor. Notes: <ul style="list-style-type: none"> The value is in electrical degrees. The electrical angle equals the mechanical angle multiplied by the number of motor pole pairs. This parameter is valid only for permanent magnet motors. 	
	0...360°	Angle offset.	1 = 1°

99 Start-up data		Language selection, motor configuration and ID run settings.	
99.01	Language	Selects the language of the control panel displays. Note: Not all languages listed below are necessarily supported.	
	English	English.	0809 hex
	Deutsch	German.	0407 hex
	Italiano	Italian.	0410 hex
	Español	Spanish.	040A hex
	Français	French.	040C hex
	Dansk	Danish.	0406 hex
	Svenska	Swedish.	041D hex
	Russki	Russian.	0419 hex
	Türkçe	Turkish.	041F hex

No.	Name/Value	Description	FbEq
99.04	Motor type	Selects the motor type. Note: This parameter cannot be changed while the drive is running.	
	AM	Asynchronous motor. Three-phase AC induction motor with squirrel cage rotor.	0
	PMSM	Permanent magnet motor. Three-phase AC synchronous motor with permanent magnet rotor and sinusoidal BackEMF voltage.	1
99.05	Motor ctrl mode	Selects the motor control mode.	
	DTC	Direct torque control. This mode is suitable for most applications. Note: Instead of direct torque control, use scalar control <ul style="list-style-type: none"> • with multimotor applications 1) if the load is not equally shared between the motors, 2) if the motors are of different sizes, or 3) if the motors are going to be changed after the motor identification (ID run), • if the nominal current of the motor is less than 1/6 of the nominal output current of the drive, • if the drive is used with no motor connected (for example, for test purposes), • if the drive runs a medium-voltage motor through a step-up transformer. 	0
	Scalar	Scalar control. This mode is suitable in special cases where DTC cannot be applied. In scalar control, the drive is controlled with a frequency reference. The outstanding motor control accuracy of DTC cannot be achieved in scalar control. Some standard features are disabled in scalar control mode. Note: Correct motor run requires that the magnetizing current of the motor does not exceed 90% of the nominal current of the inverter. See also section Scalar motor control on page 65.	1
99.06	Mot nom current	Defines the nominal motor current. Must be equal to the value on the motor rating plate. If multiple motors are connected to the drive, enter the total current of the motors. Notes: <ul style="list-style-type: none"> • Correct motor run requires that the magnetizing current of the motor does not exceed 90% of the nominal current of the drive. • This parameter cannot be changed while the drive is running. 	
	0.0 ... 6400.0 A	Nominal current of the motor. The allowable range is $1/6 \dots 2 \times I_{Hd}$ of the drive ($0 \dots 2 \times I_{Hd}$ with scalar control mode).	10 = 1 A

No.	Name/Value	Description	FbEq
99.07	Mot nom voltage	<p>Defines the nominal motor voltage as fundamental phase-to-phase rms voltage supplied to the motor at the nominal operating point. This setting must match the value on the rating plate of the motor.</p> <p>Notes:</p> <ul style="list-style-type: none"> • With permanent magnet motors, the nominal voltage is the BackEMF voltage at nominal speed of the motor. If the voltage is given as voltage per rpm, e.g. 60 V per 1000 rpm, the voltage for a nominal speed of 3000 rpm is $3 \times 60 \text{ V} = 180 \text{ V}$. Note that the nominal voltage is not equal to the equivalent DC motor voltage (EDCM) specified by some motor manufacturers. The nominal voltage can be calculated by dividing the EDCM voltage by 1.7 (or square root of 3). • The stress on the motor insulation is always dependent on the drive supply voltage. This also applies to the case where the motor voltage rating is lower than that of the drive and the supply. • This parameter cannot be changed while the drive is running. 	
	$1/6 \dots 2 \times U_N$	Nominal voltage of the motor.	$10 = 1 \text{ V}$
99.08	Mot nom freq	<p>Defines the nominal motor frequency.</p> <p>Note: This parameter cannot be changed while the drive is running.</p>	
	5.0 ... 500.0 Hz	Nominal frequency of the motor.	$10 = 1 \text{ Hz}$
99.09	Mot nom speed	<p>Defines the nominal motor speed. The setting must match the value on the rating plate of the motor.</p> <p>Note: This parameter cannot be changed while the drive is running.</p> <p>Note: For safety reasons, after ID run, the maximum and minimum speed limits (parameters 20.01 and 20.02) are automatically set to a 1.2 times bigger value than the nominal motor speed.</p>	
	0 ... 10000 rpm	Nominal speed of the motor.	$1 = 1 \text{ rpm}$
99.10	Mot nom power	<p>Defines the nominal motor power. The setting must match the value on the rating plate of the motor. If multiple motors are connected to the drive, enter the total power of the motors. The unit is selected by parameter 16.17 Power unit.</p> <p>Note: This parameter cannot be changed while the drive is running.</p>	
	0.00 ... 10000.00 kW	Nominal power of the motor.	$100 = 1 \text{ kW}$
99.11	Mot nom cosφii	<p>Defines the cosφii of the motor for a more accurate motor model. (Not applicable to permanent magnet motors.) Not obligatory; if set, should match the value on the rating plate of the motor.</p> <p>Note: This parameter cannot be changed while the drive is running.</p>	
	0.00 ... 1.00	Cosφii of the motor.	$100 = 1$
99.12	Mot nom torque	<p>Defines the nominal motor shaft torque for a more accurate motor model. Not obligatory.</p> <p>Note: This parameter cannot be changed while the drive is running.</p>	

No.	Name/Value	Description	FbEq
	0 ... 2147483.647 Nm	Nominal motor torque.	1000 = 1 N•m
99.13	IDrun mode	<p>Selects the type of the motor identification performed at the next start of the drive (for Direct Torque Control). During the identification, the drive will identify the characteristics of the motor for optimum motor control. After the ID run, the drive is stopped. Note: This parameter cannot be changed while the drive is running.</p> <p>Once the ID run is activated, it can be cancelled by stopping the drive: If ID run has already been performed once, parameter is automatically set to NO. If no ID run has been performed yet, parameter is automatically set to <i>Standstill</i>. In this case, the ID run must be performed.</p> <p>Notes:</p> <ul style="list-style-type: none"> • ID run can only be performed in local control (i.e. when drive is controlled via PC tool or control panel). • ID run cannot be performed if parameter <i>99.05 Motor ctrl mode</i> is set to <i>Scalar</i>. • ID run must be performed every time any of the motor parameters (<i>99.04, 99.06...99.12</i>) have been changed. Parameter is automatically set to <i>Standstill</i> after the motor parameters have been set. With permanent magnet motor, the motor shaft must NOT be locked and the load torque must be < 10% during the ID run (<i>Normal/Reduced/Standstill</i>). • Ensure that possible Safe torque off and emergency stop circuits are closed during ID run. • Mechanical brake is not opened by the logic for the ID run. 	
	No	No motor ID run is requested. This mode can be selected only if the ID run (Normal/Reduced/Standstill) has already been performed once.	0
	Normal	<p>Normal ID run. Guarantees the best possible control accuracy. The ID run takes about 90 seconds. This mode should be selected whenever it is possible.</p> <p>Notes:</p> <ul style="list-style-type: none"> • The driven machinery must be de-coupled from the motor with Normal ID run, if the load torque is higher than 20%, or if the machinery is not able to withstand the nominal torque transient during the ID run. • Check the direction of rotation of the motor before starting the ID run. During the run, the motor will rotate in the forward direction. <p> WARNING! The motor will run at up to approximately 50...100% of the nominal speed during the ID run. ENSURE THAT IT IS SAFE TO RUN THE MOTOR BEFORE PERFORMING THE ID RUN!</p>	1

No.	Name/Value	Description	FbEq
	Reduced	<p>Reduced ID Run. This mode should be selected instead of the Normal ID Run if</p> <ul style="list-style-type: none"> mechanical losses are higher than 20% (i.e. the motor cannot be de-coupled from the driven equipment), or if flux reduction is not allowed while the motor is running (i.e. in case of a motor with an integrated brake supplied from the motor terminals). <p>With Reduced ID run, the control in the field weakening area or at high torques is not necessarily as accurate as with the Normal ID run. Reduced ID run is completed faster than the Normal ID Run (< 90 seconds).</p> <p>Note: Check the direction of rotation of the motor before starting the ID run. During the run, the motor will rotate in the forward direction.</p> <p> WARNING! The motor will run at up to approximately 50...100% of the nominal speed during the ID run. ENSURE THAT IT IS SAFE TO RUN THE MOTOR BEFORE PERFORMING THE ID RUN!</p>	2
	Standstill	<p>Standstill ID run. The motor is injected with DC current. With an asynchronous motor, the motor shaft is not rotating (with permanent magnet motor the shaft can rotate < 0.5 revolution).</p> <p>Note: This mode should be selected only if the <i>Normal</i> or <i>Reduced</i> ID run is not possible due to the restrictions caused by the connected mechanics (e.g. with lift or crane applications).</p>	3
	Autophasing	<p>During autophasing, the start angle of the motor is determined. Note that other motor model values are not updated. See also parameter 11.07 Autophasing mode.</p> <p>Notes:</p> <ul style="list-style-type: none"> Autophasing can only be selected after the Normal/Reduced/Standstill ID run has been performed once. Autophasing is used when an absolute encoder, a resolver or an encoder with commutation signals has been added/changed to a permanent magnet motor and there is no need to perform the Normal/Reduced/Standstill ID run again. During Autophasing, the motor shaft must NOT be locked and the load torque must be < 5%. 	4
	Cur meas cal	<p>Current offset and gain measurement calibration. The calibration will be performed at next start.</p>	5



Additional parameter data

efesotomasyon.com

What this chapter contains

This chapter lists the parameters with some additional data. For parameter descriptions, see chapter [Parameters](#) on page 101.

Terms and abbreviations

Term	Definition
Actual signal	Signal measured or calculated by the drive. Usually can only be monitored but not adjusted; some counters can however be reset by entering 0.
Bit pointer	Bit pointer. A bit pointer can point to a single bit in the value of another parameter, or be fixed to 0 (C.FALSE) or 1 (C.TRUE).
enum	Enumerated list, i.e. selection list.
FbEq	Fieldbus equivalent: The scaling between the value shown on the panel and the integer used in serial communication.
INT32	32-bit integer value (31 bits + sign).
No.	Parameter number.
Pb	Packed boolean.
REAL	$\underbrace{\text{16-bit value}} \quad \underbrace{\text{16-bit value}} \quad (31 \text{ bits} + \text{sign})$ = integer value = fractional value
REAL24	$\underbrace{\text{8-bit value}} \quad \underbrace{\text{24-bit value}} \quad (31 \text{ bits} + \text{sign})$ = integer value = fractional value

Type	Data type. See enum, INT32, Bit pointer, Val pointer, Pb, REAL, REAL24, UINT32.
UINT32	32-bit unsigned integer value.
Val pointer	Value pointer. Points to the value of another parameter.

Fieldbus equivalents

Serial communication data between the fieldbus adapter and the drive is transferred in integer format. Thus, the drive actual and reference signal values must be scaled to 16/32-bit integer values. The fieldbus equivalent defines the scaling between the signal value and the integer used in serial communication.

All the read and sent values are limited to 16/32 bits.

Example: If [24.03 Maximum torq ref](#) is set from an external control system, an integer value of 10 corresponds to 1%.

Pointer parameter format in fieldbus communication

Value and bit pointer parameters are transferred between the fieldbus and drive as 32-bit integer values.

■ 32-bit integer value pointers

When a value pointer parameter is connected to the value of another parameter, the format is as follows:

	Bit			
	30...31	16...29	8...15	0...7
Name	Source type	Not in use	Group	Index
Value	1	-	1...255	1...255
Description	Value pointer is connected to parameter	-	Group of source parameter	Index of source parameter

For example, the value that should be written into parameter [33.02 Superv1 act](#) to change its value to [01.07 Dc-voltage](#) is

0100 0000 0000 0000 0000 0001 0000 0111 = 1073742087 (32-bit integer).

When a value pointer parameter is connected to an application program, the format is as follows:

	Bit		
	30...31	24...29	0...23
Name	Source type	Not in use	Address
Value	2	-	0 ... $2^{24}-1$
Description	Value pointer is connected to application program.	-	Relative address of application program variable

Note: Value pointer parameters connected to an application program are read-only via fieldbus.

■ 32-bit integer bit pointers

When a bit pointer parameter is connected to value 0 or 1, the format is as follows:

	Bit			
	30...31	16...29	1...15	0
Name	Source type	Not in use	Not in use	Value
Value	0	-	-	0...1
Description	Bit pointer is connected to 0/1.	-	-	0 = False, 1 = True

When a bit pointer parameter is connected to a bit value of another parameter, the format is as follows:

	Bit				
	30...31	24...29	16...23	8...15	0...7
Name	Source type	Not in use	Bit sel	Group	Index
Value	1	-	0...31	2...255	1...255
Description	Bit pointer is connected to signal bit value.	-	Bit selection	Group of source parameter	Index of source parameter

When a bit pointer parameter is connected to an application program, the format is as follows:

	Bit		
	30...31	24...29	0...23
Name	Source type	Bit sel	Address
Value	2	0...31	0 ... $2^{24}-1$

	Bit		
	30...31	24...29	0...23
Description	Bit pointer is connected to application program.	Bit selection	Relative address of application program variable

Note: Bit pointer parameters connected to an application program are read-only via fieldbus.

Parameter groups 1...9

No.	Name	Type	Data length	Range	Unit	Update time	Notes
01 Actual values							
01.01	Motor speed rpm	REAL	32	-30000...30000	rpm	250 µs	
01.02	Motor speed %	REAL	32	-1000...1000	%	2 ms	
01.03	Output frequency	REAL	32	-30000...30000	Hz	2 ms	
01.04	Motor current	REAL	32	0...30000	A	10 ms	
01.05	Motor current %	REAL	16	0...1000	%	2 ms	
01.06	Motor torque	REAL	16	-1600...1600	%	2 ms	
01.07	Dc-voltage	REAL	32	0...2000	V	2 ms	
01.08	Encoder1 speed	REAL	32	-32768...32768	rpm	250 µs	
01.09	Encoder1 pos	REAL24	32	0...1	rev	250 µs	
01.10	Encoder2 speed	REAL	32	-32768...32768	rpm	250 µs	
01.11	Encoder2 pos	REAL24	32	0...1	rev	250 µs	
01.12	Pos act	REAL	32	-32768...32768	rev	2 ms	
01.13	Pos 2nd enc	REAL	32	-32768...32768	rev	2 ms	
01.14	Motor speed est	REAL	32	-30000...30000	rpm	2 ms	
01.15	Temp inverter	REAL24	16	-40...160	%	2 ms	
01.16	Temp brk chopper	REAL24	16	-40...160	%	2 ms	
01.17	Motor temp1	REAL	16	-10...250	°C	10 ms	
01.18	Motor temp2	REAL	16	-10...250	°C	10 ms	
01.19	Used supply volt	REAL	16	0...1000	V	10 ms	
01.20	Brake res load	REAL24	16	0...1000	%	50 ms	
01.21	Cpu usage	UINT32	16	0...100	%	-	
01.22	Power inu out	REAL	32	-32768...32768	kW or hp	10 ms	
01.23	Motor power	REAL	32	-32768...32768	kW or hp	2 ms	
01.24	kWh inverter	INT32	32	0...2147483647	kWh	10 ms	
01.25	kWh supply	INT32	32	-2147483647 ... 2147483647	kWh	10 ms	
01.26	On-time counter	INT32	32	0...35791394.1	h	10 ms	
01.27	Run-time counter	INT32	32	0...35791394.1	h	10 ms	
01.28	Fan on-time	INT32	32	0...35791394.1	h	10 ms	
01.29	Torq nom scale	INT32	32	0...2147483.647	Nm	-	
01.30	Polepairs	INT32	16	0...1000	-	-	
01.31	Mech time const	REAL	32	0...32767	s	10 ms	
01.32	Temp phase A	REAL24	16	-40...160	%	2 ms	
01.33	Temp phase B	REAL24	16	-40...160	%	2 ms	
01.34	Temp phase C	REAL24	16	-40...160	%	2 ms	
01.35	Saved energy	INT32	32	0...2147483647	kWh	10 ms	
01.36	Saved amount	INT32	32	0...21474836.47	-	10 ms	
01.37	Saved CO2	INT32	32	0...214748364.7	t	10 ms	
01.38	Temp int board	REAL24	16	-40...160	°C	2 ms	
02 I/O values							
02.01	DI status	Pb	16	0b00000000 ... 0b11111111	-	2 ms	
02.02	RO status	Pb	16	0b00000000 ... 0b11111111	-	2 ms	
02.03	DIO status	Pb	16	0b0000000000 ... 0b1111111111	-	2 ms	
02.04	AI1	REAL	16	-11...11 V or -22...22 mA	V or mA	2 ms	

262 Additional parameter data

No.	Name	Type	Data length	Range	Unit	Update time	Notes
02.05	<i>AI1 scaled</i>	REAL	32	-32768...32768	-	2 ms	
02.06	<i>AI2</i>	REAL	16	-11...11 V or -22...22 mA	V or mA	2 ms	
02.07	<i>AI2 scaled</i>	REAL	32	-32768...32768	-	2 ms	
02.08	<i>AI3</i>	REAL	16	-22...22	mA	2 ms	
02.09	<i>AI3 scaled</i>	REAL	32	-32768...32768	-	2 ms	
02.10	<i>AI4</i>	REAL	16	-22...22	mA	2 ms	
02.11	<i>AI4 scaled</i>	REAL	32	-32768...32768	-	2 ms	
02.12	<i>AI5</i>	REAL	16	-22...22	mA	2 ms	
02.13	<i>AI5 scaled</i>	REAL	32	-32768...32768	-	2 ms	
02.14	<i>AI6</i>	REAL	16	-22...22	mA	2 ms	
02.15	<i>AI6 scaled</i>	REAL	32	-32768...32768	-	2 ms	
02.16	<i>AO1</i>	REAL	16	0 ... 22.7	mA	2 ms	
02.17	<i>AO2</i>	REAL	16	0 ... 22.7	mA	2 ms	
02.18	<i>AO3</i>	REAL	16	0 ... 22.7	mA	2 ms	
02.19	<i>AO4</i>	REAL	16	0 ... 22.7	mA	2 ms	
02.20	<i>Freq in</i>	REAL	32	-32768...32768	-	250 µs	
02.21	<i>Freq out</i>	REAL	32	0...32767	Hz	250 µs	
02.22	<i>FBA main cw</i>	Pb	32	0x00000000 ... 0xFFFFFFFF	-	500 µs	
02.24	<i>FBA main sw</i>	Pb	32	0x00000000 ... 0xFFFFFFFF	-	500 µs	
02.26	<i>FBA main ref1</i>	INT32	32	-2147483647 ... 2147483647	-	500 µs	
02.27	<i>FBA main ref2</i>	INT32	32	-2147483647 ... 2147483647	-	500 µs	
02.30	<i>D2D main cw</i>	Pb	16	0x0000...0xFFFF	-	500 µs	
02.31	<i>D2D follower cw</i>	Pb	16	0x0000...0xFFFF	-	2 ms	
02.32	<i>D2D ref1</i>	REAL	32	-2147483647 ... 2147483647	-	500 µs	
02.33	<i>D2D ref2</i>	REAL	32	-2147483647 ... 2147483647	-	2 ms	
02.34	<i>Panel ref</i>	REAL	32	-32768...32768	rpm or %	10 ms	
02.35	<i>FEN DI status</i>	Pb	16	0...0x33	-	500 µs	
02.36	<i>EFB main cw</i>	Pb	32	0x00000000 ... 0xFFFFFFFF	-	10 ms	
02.37	<i>EFB main sw</i>	Pb	32	0x00000000 ... 0xFFFFFFFF	-	10 ms	
02.38	<i>EFB main ref1</i>	INT32	32	-2147483647 ... 2147483647	-	10 ms	
02.39	<i>EFB main ref2</i>	INT32	32	-2147483647 ... 2147483647	-	10 ms	
03 Control values							
03.03	<i>SpeedRef unramp</i>	REAL	32	-30000...30000	rpm	250 µs	
03.05	<i>SpeedRef ramped</i>	REAL	32	-30000...30000	rpm	250 µs	
03.06	<i>SpeedRef used</i>	REAL	32	-30000...30000	rpm	250 µs	
03.07	<i>Speed error filt</i>	REAL	32	-30000...30000	rpm	250 µs	
03.08	<i>Acc comp torq</i>	REAL	16	-1600...1600	%	250 µs	
03.09	<i>Torq ref sp ctrl</i>	REAL	16	-1600...1600	%	250 µs	
03.11	<i>Torq ref ramped</i>	REAL	16	-1000...1000	%	250 µs	
03.12	<i>Torq ref sp lim</i>	REAL	16	-1000...1000	%	250 µs	
03.13	<i>Torq ref to TC</i>	REAL	16	-1600...1600	%	250 µs	

No.	Name	Type	Data length	Range	Unit	Update time	Notes
03.14	<i>Torq ref used</i>	REAL	16	-1600...1600	%	250 µs	
03.15	<i>Brake torq mem</i>	REAL	16	-1000...1000	%	2 ms	
03.16	<i>Brake command</i>	enum	16	0...1	-	2 ms	
03.17	<i>Flux actual</i>	REAL24	16	0...200	%	2 ms	
03.18	<i>Speed ref pot</i>	REAL	32	-30000...30000	rpm	10 ms	
03.20	<i>Max speed ref</i>	REAL	16	0...30000	rpm	2 ms	
03.21	<i>Min speed ref</i>	REAL	16	-30000...0	rpm	2 ms	
04 Appl values							
04.01	<i>Process act1</i>	REAL	32	-32768...32768	-	2 ms	
04.02	<i>Process act2</i>	REAL	32	-32768...32768	-	2 ms	
04.03	<i>Process act</i>	REAL	32	-32768...32768	-	2 ms	
04.04	<i>Process PID err</i>	REAL	32	-32768...32768	-	2 ms	
04.05	<i>Process PID out</i>	REAL	32	-32768...32768	-	2 ms	
04.06	<i>Process var1</i>	REAL	32	-32768...32768	-	10 ms	
04.07	<i>Process var2</i>	REAL	32	-32768...32768	-	10 ms	
04.08	<i>Process var3</i>	REAL	32	-32768...32768	-	10 ms	
04.09	<i>Counter ontime1</i>	UINT32	32	0...2147483647	s	10 ms	
04.10	<i>Counter ontime2</i>	UINT32	32	0...2147483647	s	10 ms	
04.11	<i>Counter edge1</i>	UINT32	32	0...2147483647	-	10 ms	
04.12	<i>Counter edge2</i>	UINT32	32	0...2147483647	-	10 ms	
04.13	<i>Counter value1</i>	UINT32	32	0...2147483647	-	10 ms	
04.14	<i>Counter value2</i>	UINT32	32	0...2147483647	-	10 ms	
06 Drive status							
06.01	<i>Status word1</i>	Pb	16	0x0000...0xFFFF	-	2 ms	
06.02	<i>Status word2</i>	Pb	16	0x0000...0xFFFF	-	2 ms	
06.03	<i>Speed ctrl stat</i>	Pb	16	0x0000...0xFFFF	-	250 µs	
06.05	<i>Limit word1</i>	Pb	16	0x0000...0xFFFF	-	250 µs	
06.07	<i>Torq lim status</i>	Pb	16	0x0000...0xFFFF	-	250 µs	
06.12	<i>Op mode ack</i>	enum	16	0...11	-	2 ms	
06.13	<i>Superv status</i>	Pb	16	0b00...0b11	-	2 ms	
06.14	<i>Timed func stat</i>	Pb	16	0b0000...0b1111	-	10 ms	
06.15	<i>Counter status</i>	Pb	16	0b000000...0b111111	-	10 ms	
08 Alarms & faults							
08.01	<i>Active fault</i>	enum	16	0...65535	-	-	
08.02	<i>Last fault</i>	enum	16	0...2147483647	-	-	
08.03	<i>Fault time hi</i>	INT32	32	$-2^{31} \dots 2^{31} - 1$	(date)	-	
08.04	<i>Fault time lo</i>	INT32	32	00:00:00 ... 24:00:00	(time)	-	
08.05	<i>Alarm word1</i>	UINT32	16	0x0000...0xFFFF	-	2 ms	
08.06	<i>Alarm word2</i>	UINT32	16	0x0000...0xFFFF	-	2 ms	
08.07	<i>Alarm word3</i>	UINT32	16	0x0000...0xFFFF	-	2 ms	
08.08	<i>Alarm word4</i>	UINT32	16	0x0000...0xFFFF	-	2 ms	
09 System info							
09.01	<i>Drive type</i>	INT32	16	0...65535	-	-	
09.02	<i>Drive rating ID</i>	INT32	16	0...65535	-	-	
09.03	<i>Firmware ID</i>	Pb	16	-	-	-	
09.04	<i>Firmware ver</i>	Pb	16	-	-	-	
09.05	<i>Firmware patch</i>	Pb	16	-	-	-	
09.10	<i>Int logic ver</i>	Pb	32	-	-	-	
09.20	<i>Option slot1</i>	INT32	16	0...21	-	-	
09.21	<i>Option slot2</i>	INT32	16	0...21	-	-	
09.22	<i>Option slot3</i>	INT32	16	0...21	-	-	

Parameter groups 10...99

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
10 Start/stop/dir						
10.01	<i>Ext1 start func</i>	enum	16	0...6	-	<i>In1</i>
10.02	<i>Ext1 start in1</i>	Bit pointer	32	-	-	<i>DI1</i>
10.03	<i>Ext1 start in2</i>	Bit pointer	32	-	-	C.FALSE
10.04	<i>Ext2 start func</i>	enum	16	0...6	-	<i>Not sel</i>
10.05	<i>Ext2 start in1</i>	Bit pointer	32	-	-	C.FALSE
10.06	<i>Ext2 start in2</i>	Bit pointer	32	-	-	C.FALSE
10.07	<i>Jog1 start</i>	Bit pointer	32	-	-	C.FALSE
10.08	<i>Jog2 start</i>	Bit pointer	32	-	-	C.FALSE
10.09	<i>Jog enable</i>	Bit pointer	32	-	-	C.FALSE
10.10	<i>Fault reset sel</i>	Bit pointer	32	-	-	<i>DI3</i>
10.11	<i>Run enable</i>	Bit pointer	32	-	-	C.TRUE
10.13	<i>Em stop off3</i>	Bit pointer	32	-	-	C.TRUE
10.15	<i>Em stop off1</i>	Bit pointer	32	-	-	C.TRUE
10.17	<i>Start enable</i>	Bit pointer	32	-	-	C.TRUE
10.19	<i>Start inhibit</i>	enum	16	0...1	-	<i>Disabled</i>
10.20	<i>Start interl func</i>	enum	16	0...1	-	<i>Off2 stop</i>
11 Start/stop mode						
11.01	<i>Start mode</i>	enum	16	0...2	-	<i>Automatic</i>
11.02	<i>Dc-magn time</i>	UINT32	16	0...10000	ms	500 ms
11.03	<i>Stop mode</i>	enum	16	1...2	-	<i>Coast</i>
11.04	<i>Dc hold speed</i>	REAL	16	0...1000	rpm	5.0 rpm
11.05	<i>Dc hold curr ref</i>	UINT32	16	0...100	%	30%
11.06	<i>Dc hold</i>	enum	16	0...1	-	<i>Disabled</i>
11.07	<i>Autophasing mode</i>	enum	16	0...2	-	<i>Turning</i>
12 Operating mode						
12.01	<i>Ext1/Ext2 sel</i>	Bit pointer	32	-	-	C.FALSE
12.03	<i>Ext1 ctrl mode</i>	enum	16	1...5	-	<i>Speed</i>
12.05	<i>Ext2 ctrl mode</i>	enum	16	1...5	-	<i>Speed</i>
12.07	<i>Local ctrl mode</i>	enum	16	1...2	-	<i>Speed</i>
13 Analogue inputs						
13.01	<i>A11 filt time</i>	REAL	16	0...30	s	0.100 s
13.02	<i>A11 max</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	10.000 V
13.03	<i>A11 min</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	-10.000 V
13.04	<i>A11 max scale</i>	REAL	32	-32768...32768	-	1500.000
13.05	<i>A11 min scale</i>	REAL	32	-32768...32768	-	-1500.000
13.06	<i>A12 filt time</i>	REAL	16	0...30	s	0.100 s
13.07	<i>A12 max</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	10.000 V
13.08	<i>A12 min</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	-10.000 V

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
13.09	<i>AI2 max scale</i>	REAL	32	-32768...32768	-	100.000
13.10	<i>AI2 min scale</i>	REAL	32	-32768...32768	-	-100.000
13.11	<i>AI3 filt time</i>	REAL	16	0...30	s	0.100 s
13.12	<i>AI3 max</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	22.000 mA
13.13	<i>AI3 min</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	4.000 mA
13.14	<i>AI3 max scale</i>	REAL	32	-32768...32768	-	1500.000
13.15	<i>AI3 min scale</i>	REAL	32	-32768...32768	-	0.000
13.16	<i>AI4 filt time</i>	REAL	16	0...30	s	0.100 s
13.17	<i>AI4 max</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	22.000 mA
13.18	<i>AI4 min</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	4.000 mA
13.19	<i>AI4 max scale</i>	REAL	32	-32768...32768	-	1500.000
13.20	<i>AI4 min scale</i>	REAL	32	-32768...32768	-	0.000
13.21	<i>AI5 filt time</i>	REAL	16	0...30	s	0.100 s
13.22	<i>AI5 max</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	22.000 mA
13.23	<i>AI5 min</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	4.000 mA
13.24	<i>AI5 max scale</i>	REAL	32	-32768...32768	-	1500.000
13.25	<i>AI5 min scale</i>	REAL	32	-32768...32768	-	0.000
13.26	<i>AI6 filt time</i>	REAL	16	0...30	s	0.100 s
13.27	<i>AI6 max</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	22.000 mA
13.28	<i>AI6 min</i>	REAL	16	-22...22 mA or -11...11 V	mA or V	4.000 mA
13.29	<i>AI6 max scale</i>	REAL	32	-32768...32768	-	1500.000
13.30	<i>AI6 min scale</i>	REAL	32	-32768...32768	-	0.000
13.31	<i>AI tune</i>	enum	16	0...4	-	<i>No action</i>
13.32	<i>AI superv func</i>	enum	16	0...3	-	<i>No</i>
13.33	<i>AI superv cw</i>	UINT32	32	0b0000...0b1111	-	0b0000
14 Digital I/O						
14.01	<i>DI invert mask</i>	Pb	16	0b00000000 ... 0b1111111111	-	0b00000000
14.02	<i>DIO1 conf</i>	enum	16	0...2	-	<i>Output</i>
14.03	<i>DIO1 out src</i>	Bit pointer	32	-	-	<i>Ready relay</i>
14.04	<i>DIO1 Ton</i>	UINT32	16	0...3000	s	0.0 s
14.05	<i>DIO1 Toff</i>	UINT32	16	0...3000	s	0.0 s
14.06	<i>DIO2 conf</i>	enum	16	0...3	-	<i>Output</i>
14.07	<i>DIO2 out src</i>	Bit pointer	32	-	-	<i>RunningRelay</i>
14.08	<i>DIO2 Ton</i>	UINT32	16	0...3000	s	0.0 s
14.09	<i>DIO2 Toff</i>	UINT32	16	0...3000	s	0.0 s
14.10	<i>DIO3 conf</i>	enum	16	0...1	-	<i>Output</i>
14.11	<i>DIO3 out src</i>	Bit pointer	32	-	-	<i>Fault(-1)</i>
14.14	<i>DIO4 conf</i>	enum	16	0...1	-	<i>Output</i>
14.15	<i>DIO4 out src</i>	Bit pointer	32	-	-	<i>Ready relay</i>
14.18	<i>DIO5 conf</i>	enum	16	0...1	-	<i>Output</i>

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
14.19	<i>DIO5 out src</i>	Bit pointer	32	-	-	<i>Ref running</i>
14.22	<i>DIO6 conf</i>	enum	16	0...1	-	<i>Output</i>
14.23	<i>DIO6 out src</i>	Bit pointer	32	-	-	<i>Fault</i>
14.26	<i>DIO7 conf</i>	enum	16	0...1	-	<i>Output</i>
14.27	<i>DIO7 out src</i>	Bit pointer	32	-	-	<i>Alarm</i>
14.30	<i>DIO8 conf</i>	enum	16	0...1	-	<i>Output</i>
14.31	<i>DIO8 out src</i>	Bit pointer	32	-	-	<i>Ext2 active</i>
14.34	<i>DIO9 conf</i>	enum	16	0...1	-	<i>Output</i>
14.35	<i>DIO9 out src</i>	Bit pointer	32	-	-	<i>At setpoint</i>
14.38	<i>DIO10 conf</i>	enum	16	0...1	-	<i>Output</i>
14.39	<i>DIO10 out src</i>	Bit pointer	32	-	-	<i>Zero speed</i>
14.42	<i>RO1 src</i>	Bit pointer	32	-	-	<i>Ready relay</i>
14.43	<i>RO1 Ton</i>	UINT32	16	0...3000	s	0.0 s
14.44	<i>RO1 Toff</i>	UINT32	16	0...3000	s	0.0 s
14.45	<i>RO2 src</i>	Bit pointer	32	-	-	<i>RunningRelay</i>
14.48	<i>RO3 src</i>	Bit pointer	32	-	-	<i>Fault(-1)</i>
14.51	<i>RO4 src</i>	Bit pointer	32	-	-	P.06.02.02
14.54	<i>RO5 src</i>	Bit pointer	32	-	-	P.06.02.04
14.57	<i>Freq in max</i>	REAL	16	3...32768	Hz	1000 Hz
14.58	<i>Freq in min</i>	REAL	16	3...32768	Hz	3 Hz
14.59	<i>Freq in max scal</i>	REAL	16	-32768...32768	-	1500
14.60	<i>Freq in min scal</i>	REAL	16	-32768... 32768	-	0
14.61	<i>Freq out src</i>	Val pointer	32	-	-	P.01.01
14.62	<i>Freq out max src</i>	REAL	16	0...32768	-	1500
14.63	<i>Freq out min src</i>	REAL	16	0...32768	-	0
14.64	<i>Freq out max sca</i>	REAL	16	3...32768	Hz	1000 Hz
14.65	<i>Freq out min sca</i>	REAL	16	3...32768	Hz	3 Hz
14.66	<i>RO6 src</i>	Bit pointer	32	-	-	C.FALSE
14.69	<i>RO7 src</i>	Bit pointer	32	-	-	C.FALSE
14.72	<i>DIO invert mask</i>	Pb	16	0b0000000000 ... 0b1111111111	-	0b0000000000
15 Analogue outputs						
15.01	<i>AO1 src</i>	Val pointer	32	-	-	<i>Current %</i>
15.02	<i>AO1 filt time</i>	REAL	16	0...30	s	0.100 s
15.03	<i>AO1 out max</i>	REAL	16	0 ... 22.7	mA	20.000 mA
15.04	<i>AO1 out min</i>	REAL	16	0 ... 22.7	mA	4.000 mA
15.05	<i>AO1 src max</i>	REAL	32	-32768...32768	-	100.000
15.06	<i>AO1 src min</i>	REAL	32	-32768...32768	-	0.000
15.07	<i>AO2 src</i>	Val pointer	32	-	-	<i>Speed %</i>
15.08	<i>AO2 filt time</i>	REAL	16	0...30	s	0.100 s
15.09	<i>AO2 out max</i>	REAL	16	0 ... 22.7	mA	20.000 mA

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
15.10	<i>AO2 out min</i>	REAL	16	0 ... 22.7	mA	4.000 mA
15.11	<i>AO2 src max</i>	REAL	32	-32768...32768	-	100.000
15.12	<i>AO2 src min</i>	REAL	32	-32768...32768	-	-100.000
15.13	<i>AO3 src</i>	Val pointer	32	-	-	<i>Frequency</i>
15.14	<i>AO3 filt time</i>	REAL	16	0...30	s	0.100 s
15.15	<i>AO3 out max</i>	REAL	16	0 ... 22.7	mA	22.000 mA
15.16	<i>AO3 out min</i>	REAL	16	0 ... 22.7	mA	4.000 mA
15.17	<i>AO3 src max</i>	REAL	32	-32768...32768	-	50.000
15.18	<i>AO3 src min</i>	REAL	32	-32768...32768	-	0.000
15.19	<i>AO4 src</i>	Val pointer	32	-	-	<i>Frequency</i>
15.20	<i>AO4 filt time</i>	REAL	16	0...30	s	0.100 s
15.21	<i>AO4 out max</i>	REAL	16	0 ... 22.7	mA	22.000 mA
15.22	<i>AO4 out min</i>	REAL	16	0 ... 22.7	mA	4.000 mA
15.23	<i>AO4 src max</i>	REAL	32	-32768...32768	-	50.000
15.24	<i>AO4 src min</i>	REAL	32	-32768...32768	-	0.000
15.25	<i>AO ctrl word</i>	UINT32	32	0b0000...0b1111	-	0b0000
16 System						
16.01	<i>Local lock</i>	Bit pointer	32	-	-	C.FALSE
16.02	<i>Parameter lock</i>	enum	16	0...2	-	<i>Open</i>
16.03	<i>Pass code</i>	INT32	32	0...2147483647	-	0
16.04	<i>Param restore</i>	enum	16	0...2	-	<i>Done</i>
16.07	<i>Param save</i>	enum	16	0...1	-	<i>Done</i>
16.09	<i>User set sel</i>	enum	32	1...10	-	<i>No request</i>
16.10	<i>User set log</i>	Pb	32	0...1024	-	<i>N/A</i>
16.11	<i>User IO sel lo</i>	Bit pointer	32	-	-	C.FALSE
16.12	<i>User IO sel hi</i>	Bit pointer	32	-	-	C.FALSE
16.14	<i>Reset ChgParLog</i>	enum	16	0...1	-	<i>Done</i>
16.15	<i>Menu set sel</i>	enum	16	0...2	-	<i>No request</i>
16.16	<i>Menu set active</i>	enum	16	0...2	-	<i>Short menu</i>
16.17	<i>Power unit</i>	enum	16	0...1	-	<i>kW</i>
19 Speed calculation						
19.01	<i>Speed scaling</i>	REAL	16	0...30000	rpm	1500 rpm
19.02	<i>Speed fb sel</i>	enum	16	0...2	-	<i>Estimated</i>
19.03	<i>MotorSpeed filt</i>	REAL	32	0...10000	ms	8.000 ms
19.06	<i>Zero speed limit</i>	REAL	32	0...30000	rpm	30.00 rpm
19.07	<i>Zero speed delay</i>	UINT32	16	0...30000	ms	0 ms
19.08	<i>Above speed lim</i>	REAL	16	0...30000	rpm	0 rpm
19.09	<i>Speed TripMargin</i>	REAL	32	0...10000	rpm	500.0 rpm
19.10	<i>Speed window</i>	REAL	16	0...30000	rpm	100 rpm
19.13	<i>Speed fbk fault</i>	enum	16	0...2	-	<i>Fault</i>

268 *Additional parameter data*

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
20 Limits						
20.01	<i>Maximum speed</i>	REAL	32	0...30000	rpm	1500 rpm
20.02	<i>Minimum speed</i>	REAL	32	-30000...0	rpm	-1500 rpm
20.03	<i>Pos speed ena</i>	Bit pointer	32	-	-	C.TRUE
20.04	<i>Neg speed ena</i>	Bit pointer	32	-	-	C.TRUE
20.05	<i>Maximum current</i>	REAL	32	0...30000	A	0.00 A
20.06	<i>Torq lim sel</i>	Bit pointer	32	-	-	C.FALSE
20.07	<i>Maximum torque1</i>	REAL	16	0...1600	%	300.0%
20.08	<i>Minimum torque1</i>	REAL	16	-1600...0	%	-300.0%
20.09	<i>Maximum torque2</i>	REAL	16	-	-	<i>Max torque1</i>
20.10	<i>Minimum torque2</i>	REAL	16	-	-	<i>Min torque1</i>
20.12	<i>P motoring lim</i>	REAL	16	0...1600	%	300.0%
20.13	<i>P generating lim</i>	REAL	16	0...1600	%	300.0%
21 Speed ref						
21.01	<i>Speed ref1 sel</i>	Val pointer	32	-	-	<i>AI1 scaled</i>
21.02	<i>Speed ref2 sel</i>	Val pointer	32	-	-	<i>Zero</i>
21.03	<i>Speed ref1 func</i>	enum	16	0...5	-	<i>Ref1</i>
21.04	<i>Speed ref1/2 sel</i>	Bit pointer	32	-	-	C.FALSE
21.05	<i>Speed share</i>	REAL	16	-8...8	-	1.000
21.07	<i>Speed ref jog1</i>	REAL	16	-30000...30000	rpm	0 rpm
21.08	<i>Speed ref jog2</i>	REAL	16	-30000...30000	rpm	0 rpm
21.09	<i>SpeedRef min abs</i>	REAL	16	0...30000	rpm	0 rpm
21.10	<i>Mot pot func</i>	enum	16	0...1	-	<i>Reset</i>
21.11	<i>Mot pot up</i>	Bit pointer	32	-	-	<i>DI5</i>
21.12	<i>Mot pot down</i>	Bit pointer	32	-	-	<i>DI6</i>
22 Speed ref ramp						
22.01	<i>Acc/Dec sel</i>	Bit pointer	32	-	-	C.FALSE
22.02	<i>Acc time1</i>	REAL	32	0...1800	s	20.000 s
22.03	<i>Dec time1</i>	REAL	32	0...1800	s	20.000 s
22.04	<i>Acc time2</i>	REAL	32	0...1800	s	60.000 s
22.05	<i>Dec time2</i>	REAL	32	0...1800	s	60.000 s
22.06	<i>Shape time acc1</i>	REAL	32	0...1000	s	0.100 s
22.07	<i>Shape time acc2</i>	REAL	32	0...1000	s	0.100 s
22.08	<i>Shape time dec1</i>	REAL	32	0...1000	s	0.100 s
22.09	<i>Shape time dec2</i>	REAL	32	0...1000	s	0.100 s
22.10	<i>Acc time jogging</i>	REAL	32	0...1800	s	0.000 s
22.11	<i>Dec time jogging</i>	REAL	32	0...1800	s	0.000 s
22.12	<i>Em stop time</i>	REAL	32	0...1800	s	3.000 s
23 Speed ctrl						
23.01	<i>Proport gain</i>	REAL	16	0...200	-	10.00

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
23.02	<i>Integration time</i>	REAL	32	0...600	s	0.500 s
23.03	<i>Derivation time</i>	REAL	16	0...10	s	0.000 s
23.04	<i>Deriv filt time</i>	REAL	16	0...1000	ms	8.0 ms
23.05	<i>Acc comp DerTime</i>	REAL	32	0...600	s	0.00 s
23.06	<i>Acc comp Ftime</i>	REAL	16	0...1000	ms	8.0 ms
23.07	<i>Speed err Ftime</i>	REAL	16	0...1000	ms	0.0 ms
23.08	<i>Speed additive</i>	Val pointer	32	-	-	<i>Zero</i>
23.09	<i>Max torq sp ctrl</i>	REAL	16	-1600...1600	%	300.0%
23.10	<i>Min torq sp ctrl</i>	REAL	16	-1600...1600	%	-300.0%
23.11	<i>SpeedErr winFunc</i>	enum	16	0...2	-	<i>Disabled</i>
23.12	<i>SpeedErr win hi</i>	REAL	16	0...3000	rpm	0 rpm
23.13	<i>SpeedErr win lo</i>	REAL	16	0...3000	rpm	0 rpm
23.14	<i>Drooping rate</i>	REAL	16	0...100	%	0.00%
23.15	<i>PI adapt max sp</i>	REAL	16	0...30000	rpm	0 rpm
23.16	<i>PI adapt min sp</i>	REAL	16	0...30000	rpm	0 rpm
23.17	<i>Pcoef at min sp</i>	REAL	16	0...10	-	1.000
23.18	<i>Icoef at min sp</i>	REAL	16	0...10	-	1.000
23.20	<i>PI tune mode</i>	enum	16	0...4	-	<i>Done</i>
23.21	<i>Tune bandwidth</i>	REAL	16	0...2000	Hz	100.00 Hz
23.22	<i>Tune damping</i>	REAL	16	0...200	-	1.5
24 Torque ref						
24.01	<i>Torq ref1 sel</i>	Val pointer	32	-	-	<i>A12 scaled</i>
24.02	<i>Torq ref add sel</i>	Val pointer	32	-	-	<i>Zero</i>
24.03	<i>Maximum torq ref</i>	REAL	16	0...1000	%	300.0%
24.04	<i>Minimum torq ref</i>	REAL	16	-1000...0	%	-300.0%
24.05	<i>Load share</i>	REAL	16	-8...8	-	1.000
24.06	<i>Torq ramp up</i>	UINT32	32	0...60	s	0.000 s
24.07	<i>Torq ramp down</i>	UINT32	32	0...60	s	0.000 s
25 Critical speed						
25.01	<i>Crit speed sel</i>	enum	16	0...1	-	<i>Disable</i>
25.02	<i>Crit speed1 lo</i>	REAL	16	-30000...30000	rpm	0 rpm
25.03	<i>Crit speed1 hi</i>	REAL	16	-30000...30000	rpm	0 rpm
25.04	<i>Crit speed2 lo</i>	REAL	16	-30000...30000	rpm	0 rpm
25.05	<i>Crit speed2 hi</i>	REAL	16	-30000...30000	rpm	0 rpm
25.06	<i>Crit speed3 lo</i>	REAL	16	-30000...30000	rpm	0 rpm
25.07	<i>Crit speed3 hi</i>	REAL	16	-30000...30000	rpm	0 rpm
26 Constant speeds						
26.01	<i>Const speed func</i>	Pb	16	0b00...0b11	-	0b00
26.02	<i>Const speed sel1</i>	Bit pointer	32	-	-	C.FALSE
26.03	<i>Const speed sel2</i>	Bit pointer	32	-	-	C.FALSE

270 Additional parameter data

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
26.04	<i>Const speed sel3</i>	Bit pointer	32	-	-	C.FALSE
26.06	<i>Const speed1</i>	REAL	16	-30000...30000	rpm	0 rpm
26.07	<i>Const speed2</i>	REAL	16	-30000...30000	rpm	0 rpm
26.08	<i>Const speed3</i>	REAL	16	-30000...30000	rpm	0 rpm
26.09	<i>Const speed4</i>	REAL	16	-30000...30000	rpm	0 rpm
26.10	<i>Const speed5</i>	REAL	16	-30000...30000	rpm	0 rpm
26.11	<i>Const speed6</i>	REAL	16	-30000...30000	rpm	0 rpm
26.12	<i>Const speed7</i>	REAL	16	-30000...30000	rpm	0 rpm
27 Process PID						
27.01	<i>PID setpoint sel</i>	Val pointer	32	-	-	<i>AI1 scaled</i>
27.02	<i>PID fbk func</i>	enum	16	0...8	-	<i>Act1</i>
27.03	<i>PID fbk1 src</i>	Val pointer	32	-	-	<i>AI2 scaled</i>
27.04	<i>PID fbk2 src</i>	Val pointer	32	-	-	<i>AI2 scaled</i>
27.05	<i>PID fbk1 max</i>	REAL	32	-32768...32768	-	100.00
27.06	<i>PID fbk1 min</i>	REAL	32	-32768...32768	-	-100.00
27.07	<i>PID fbk2 max</i>	REAL	32	-32768...32768	-	100.00
27.08	<i>PID fbk2 min</i>	REAL	32	-32768...32768	-	-100.00
27.09	<i>PID fbk gain</i>	REAL	16	-32.768 ... 32.767	-	1.000
27.10	<i>PID fbk ftime</i>	REAL	16	0...30	s	0.040 s
27.12	<i>PID gain</i>	REAL	16	0...100	-	1.00
27.13	<i>PID integ time</i>	REAL	16	0...320	s	60.00 s
27.14	<i>PID deriv time</i>	REAL	16	0...10	s	0.00 s
27.15	<i>PID deriv filter</i>	REAL	16	0...10	s	1.00 s
27.16	<i>PID error inv</i>	Bit pointer	32	-	-	C.FALSE
27.17	<i>PID mode</i>	enum	16	0...2	-	<i>Direct</i>
27.18	<i>PID maximum</i>	REAL	32	-32768...32768	-	100.0
27.19	<i>PID minimum</i>	REAL	32	-32768...32768	-	-100.0
27.22	<i>Sleep mode</i>	enum	16	0...2	-	<i>No</i>
27.23	<i>Sleep level</i>	REAL	32	-32768...32768	-	0.0
27.24	<i>Sleep delay</i>	UINT32	32	0...360	s	0.0 s
27.25	<i>Wake up level</i>	REAL	32	0...32768	-	0.0
27.26	<i>Wake up delay</i>	UINT32	32	0...360	s	0.0 s
27.27	<i>Sleep ena</i>	Bit pointer	32	-	-	C.FALSE
30 Fault functions						
30.01	<i>External fault</i>	Bit pointer	32	-	-	C.TRUE
30.02	<i>Speed ref safe</i>	REAL	16	-30000...30000	rpm	0 rpm
30.03	<i>Local ctrl loss</i>	enum	16	0...3	-	<i>Fault</i>
30.04	<i>Mot phase loss</i>	enum	16	0...1	-	<i>Fault</i>
30.05	<i>Earth fault</i>	enum	16	0...2	-	<i>Fault</i>
30.06	<i>Suppl phs loss</i>	enum	16	0...1	-	<i>Fault</i>

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
30.07	<i>Sto diagnostic</i>	enum	16	1...4	-	<i>Fault</i>
30.08	<i>Cross connection</i>	enum	16	0...1	-	<i>Fault</i>
30.09	<i>Stall function</i>	Pb	16	0b000...0b111	-	0b111
30.10	<i>Stall curr lim</i>	REAL	16	0...1600	%	200.0%
30.11	<i>Stall freq hi</i>	REAL	16	0.5 ... 1000	Hz	15.0 Hz
30.12	<i>Stall time</i>	UINT32	16	0...3600	s	20 s
31 Motor therm prot						
31.01	<i>Mot temp1 prot</i>	enum	16	0...2	-	<i>No</i>
31.02	<i>Mot temp1 src</i>	enum	16	0...12	-	<i>Estimated</i>
31.03	<i>Mot temp1 almLim</i>	INT32	16	0...200	°C	90 °C
31.04	<i>Mot temp1 fitLim</i>	INT32	16	0...200	°C	110 °C
31.05	<i>Mot temp2 prot</i>	enum	16	0...2	-	<i>No</i>
31.06	<i>Mot temp2 src</i>	enum	16	0...12	-	<i>Estimated</i>
31.07	<i>Mot temp2 almLim</i>	INT32	16	0...200	°C	90 °C
31.08	<i>Mot temp2 fitLim</i>	INT32	16	0...200	°C	110 °C
31.09	<i>Mot ambient temp</i>	INT32	16	-60...100	°C	20 °C
31.10	<i>Mot load curve</i>	INT32	16	50...150	%	100%
31.11	<i>Zero speed load</i>	INT32	16	50...150	%	100%
31.12	<i>Break point</i>	INT32	16	0.01...500	Hz	45.00 Hz
31.13	<i>Mot nom tempRise</i>	INT32	16	0...300	°C	80 °C
31.14	<i>Mot therm time</i>	INT32	16	100...10000	s	256 s
32 Automatic reset						
32.01	<i>Autoreset sel</i>	Pb	16	0b000000...0b111111	-	0b000000
32.02	<i>Number of trials</i>	UINT32	16	0...5	-	0
32.03	<i>Trial time</i>	UINT32	16	1...600	s	30.0 s
32.04	<i>Delay time</i>	UINT32	16	0...120	s	0.0 s
33 Supervision						
33.01	<i>Superv1 func</i>	enum	16	0...4	-	<i>Disabled</i>
33.02	<i>Superv1 act</i>	Val pointer	32	-	-	<i>Speed rpm</i>
33.03	<i>Superv1 hi</i>	REAL	32	-32768...32768	-	0.00
33.04	<i>Superv1 lo</i>	REAL	32	-32768...32768	-	0.00
33.05	<i>Superv2 func</i>	enum	16	0...4	-	<i>Disabled</i>
33.06	<i>Superv2 act</i>	Val pointer	32	-	-	<i>Current</i>
33.07	<i>Superv2 hi</i>	REAL	32	-32768...32768	-	0.00
33.08	<i>Superv2 lo</i>	REAL	32	-32768...32768	-	0.00
33.09	<i>Superv3 func</i>	enum	16	0...4	-	<i>Disabled</i>
33.10	<i>Superv3 act</i>	Val pointer	32	-	-	<i>Torque</i>
33.11	<i>Superv3 hi</i>	REAL	32	-32768...32768	-	0.00
33.12	<i>Superv3 lo</i>	REAL	32	-32768...32768	-	0.00

272 Additional parameter data

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
34 User load curve						
34.01	<i>Overload func</i>	Pb	16	0b000000...0b111111	-	0b000000
34.02	<i>Underload func</i>	Pb	16	0b0000...0b1111	-	0b0000
34.03	<i>Load freq1</i>	REAL	16	1...500	Hz	5 Hz
34.04	<i>Load freq2</i>	REAL	16	1...500	Hz	25 Hz
34.05	<i>Load freq3</i>	REAL	16	1...500	Hz	43 Hz
34.06	<i>Load freq4</i>	REAL	16	1...500	Hz	50 Hz
34.07	<i>Load freq5</i>	REAL	16	1...500	Hz	500 Hz
34.08	<i>Load low lim1</i>	REAL	16	0...1600	%	10%
34.09	<i>Load low lim2</i>	REAL	16	0...1600	%	15%
34.10	<i>Load low lim3</i>	REAL	16	0...1600	%	25%
34.11	<i>Load low lim4</i>	REAL	16	0...1600	%	30%
34.12	<i>Load low lim5</i>	REAL	16	0...1600	%	30%
34.13	<i>Load high lim1</i>	REAL	16	0...1600	%	300%
34.14	<i>Load high lim2</i>	REAL	16	0...1600	%	300%
34.15	<i>Load high lim3</i>	REAL	16	0...1600	%	300%
34.16	<i>Load high lim4</i>	REAL	16	0...1600	%	300%
34.17	<i>Load high lim5</i>	REAL	16	0...1600	%	300%
34.18	<i>Load integ time</i>	UINT32	16	0...10000	s	100 s
34.19	<i>Load cool time</i>	UINT32	16	0...10000	s	20 s
34.20	<i>Underload time</i>	UINT32	16	0...10000	s	10 s
35 Process variable						
35.01	<i>Signal1 param</i>	Val pointer	32	-	-	<i>Speed %</i>
35.02	<i>Signal1 max</i>	REAL	32	-32768...32768	-	300.000
35.03	<i>Signal1 min</i>	REAL	32	-32768...32768	-	-300.000
35.04	<i>Proc var1 dispf</i>	enum	16	0...5	-	3
35.05	<i>Proc var1 unit</i>	enum	16	0...98	-	4
35.06	<i>Proc var1 max</i>	REAL	32	-32768...32768	-	300.000
35.07	<i>Proc var1 min</i>	REAL	32	-32768...32768	-	-300.000
35.08	<i>Signal2 param</i>	Val pointer	32	-	-	<i>Current %</i>
35.09	<i>Signal2 max</i>	REAL	32	-32768...32768	-	300.000
35.10	<i>Signal2 min</i>	REAL	32	-32768...32768	-	-300.000
35.11	<i>Proc var2 dispf</i>	enum	16	0...5	-	3
35.12	<i>Proc var2 unit</i>	enum	16	0...98	-	4
35.13	<i>Proc var2 max</i>	REAL	32	-32768...32768	-	300.000
35.14	<i>Proc var2 min</i>	REAL	32	-32768...32768	-	-300.000
35.15	<i>Signal3 param</i>	Val pointer	32	-	-	<i>Torque</i>
35.16	<i>Signal3 max</i>	REAL	32	-32768...32768	-	300.000
35.17	<i>Signal3 min</i>	REAL	32	-32768...32768	-	-300.000
35.18	<i>Proc var3 dispf</i>	enum	16	0...5	-	3

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
35.19	<i>Proc var3 unit</i>	enum	16	0...98	-	4
35.20	<i>Proc var3 max</i>	REAL	32	-32768...32768	-	300.000
35.21	<i>Proc var3 min</i>	REAL	32	-32768...32768	-	-300.000
36 Timed functions						
36.01	<i>Timers enable</i>	Bit pointer	32	-	-	C.FALSE
36.02	<i>Timers mode</i>	Pb	16	0b0000...0b1111	-	0b0000
36.03	<i>Start time1</i>	UINT32	32	00:00:00 ... 24:00:00	-	00:00:00
36.04	<i>Stop time1</i>	UINT32	32	00:00:00 ... 24:00:00	-	00:00:00
36.05	<i>Start day1</i>	enum	16	1...7	-	<i>Monday</i>
36.06	<i>Stop day1</i>	enum	16	1...7	-	<i>Monday</i>
36.07	<i>Start time2</i>	UINT32	32	00:00:00 ... 24:00:00	-	00:00:00
36.08	<i>Stop time2</i>	UINT32	32	00:00:00 ... 24:00:00	-	00:00:00
36.09	<i>Start day2</i>	enum	16	1...7	-	<i>Monday</i>
36.10	<i>Stop day2</i>	enum	16	1...7	-	<i>Monday</i>
36.11	<i>Start time3</i>	UINT32	32	00:00:00 ... 24:00:00	-	00:00:00
36.12	<i>Stop time3</i>	UINT32	32	00:00:00 ... 24:00:00	-	00:00:00
36.13	<i>Start day3</i>	enum	16	1...7	-	<i>Monday</i>
36.14	<i>Stop day3</i>	enum	16	1...7	-	<i>Monday</i>
36.15	<i>Start time4</i>	UINT32	32	00:00:00 ... 24:00:00	-	00:00:00
36.16	<i>Stop time4</i>	UINT32	32	00:00:00 ... 24:00:00	-	00:00:00
36.17	<i>Start day4</i>	enum	16	1...7	-	<i>Monday</i>
36.18	<i>Stop day4</i>	enum	16	1...7	-	<i>Monday</i>
36.19	<i>Boost signal</i>	Bit pointer	32	-	-	C.FALSE
36.20	<i>Boost time</i>	UINT32	32	00:00:00 ... 24:00:00	-	00:00:00
36.21	<i>Timed func1</i>	Pb	16	0b00000...0b11111	-	0b00000
36.22	<i>Timed func2</i>	Pb	16	0b00000...0b11111	-	0b00000
36.23	<i>Timed func3</i>	Pb	16	0b00000...0b11111	-	0b00000
36.24	<i>Timed func4</i>	Pb	16	0b00000...0b11111	-	0b00000
38 Flux ref						
38.01	<i>Flux ref</i>	REAL	16	0...200	%	100%
38.03	<i>U/f curve func</i>	enum	16	0...2	-	<i>Linear</i>
38.04	<i>U/f curve freq1</i>	REAL	16	1...500	%	10%
38.05	<i>U/f curve freq2</i>	REAL	16	1...500	%	30%
38.06	<i>U/f curve freq3</i>	REAL	16	1...500	%	50%
38.07	<i>U/f curve freq4</i>	REAL	16	1...500	%	70%
38.08	<i>U/f curve freq5</i>	REAL	16	1...500	%	90%
38.09	<i>U/f curve volt1</i>	REAL	16	0...200	%	20%
38.10	<i>U/f curve volt2</i>	REAL	16	0...200	%	40%
38.11	<i>U/f curve volt3</i>	REAL	16	0...200	%	60%
38.12	<i>U/f curve volt4</i>	REAL	16	0...200	%	80%

274 Additional parameter data

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
38.13	<i>U/f curve volt5</i>	REAL	16	0...200	%	100%
38.16	<i>Flux ref pointer</i>	Val pointer	32	-	-	P.38.01
40 Motor control						
40.01	<i>Motor noise</i>	enum	16	0...2	-	<i>Cyclic</i>
40.03	<i>Slip gain</i>	REAL24	32	0...200	%	100%
40.04	<i>Voltage reserve</i>	REAL24	32	-4...50	%	-1%
40.06	<i>Force open loop</i>	enum	16	0...1	-	<i>False</i>
40.07	<i>IR-compensation</i>	REAL24	32	0...50	%	0.00%
40.10	<i>Flux braking</i>	enum	16	0...2	-	<i>Disabled</i>
42 Mech brake ctrl						
42.01	<i>Brake ctrl</i>	enum	16	0...2	-	<i>No</i>
42.02	<i>Brake acknowl</i>	Bit pointer	32	-	-	C.FALSE
42.03	<i>Open delay</i>	UINT32	16	0...5	s	0.00 s
42.04	<i>Close delay</i>	UINT32	16	0...60	s	0.00 s
42.05	<i>Close speed</i>	REAL	16	0...1000	rpm	100.0 rpm
42.06	<i>Close cmd delay</i>	UINT32	16	0...10	s	0.00 s
42.07	<i>Reopen delay</i>	UINT32	16	0...10	s	0.00 s
42.08	<i>Brake open torq</i>	REAL	16	-1000...1000	%	0.0%
42.09	<i>Open torq src</i>	Val pointer	32	-	-	<i>P.42.08</i>
42.10	<i>Brake close req</i>	Bit pointer	32	-	-	C.FALSE
42.11	<i>Brake hold open</i>	Bit pointer	32	-	-	C.FALSE
42.12	<i>Brake fault func</i>	enum	16	0...2	-	<i>Fault</i>
42.13	<i>Close flt delay</i>	UINT32	16	0...600	s	0.00 s
42.14	<i>Extend run time</i>	UINT32	16	0...3600	s	0.00 s
44 Maintenance						
44.01	<i>Ontime1 func</i>	Pb	16	0b00...0b11	-	0b01
44.02	<i>Ontime1 src</i>	Bit pointer	32	-	-	<i>Running</i>
44.03	<i>Ontime1 limit</i>	UINT32	32	0...2147483647	s	36000000 s
44.04	<i>Ontime1 alm sel</i>	enum	16	0...6	-	<i>Mot bearing</i>
44.05	<i>Ontime2 func</i>	Pb	16	0b00...0b11	-	0b01
44.06	<i>Ontime2 src</i>	Bit pointer	32	-	-	<i>Charged</i>
44.07	<i>Ontime2 limit</i>	UINT32	32	0...2147483647	s	15768000 s
44.08	<i>Ontime2 alm sel</i>	enum	16	0...6	-	<i>Device clean</i>
44.09	<i>Edge count1 func</i>	Pb	16	0b00...0b11	-	0b01
44.10	<i>Edge count1 src</i>	Bit pointer	32	-	-	<i>Charged</i>
44.11	<i>Edge count1 lim</i>	UINT32	32	0...2147483647	-	5000
44.12	<i>Edge count1 div</i>	UINT32	32	0...2147483647	-	1
44.13	<i>Edg cnt1 alm sel</i>	enum	16	0...5	-	<i>Dc-charge</i>
44.14	<i>Edge count2 func</i>	Pb	16	0b00...0b11	-	0b01
44.15	<i>Edge count2 src</i>	Bit pointer	32	-	-	<i>RO1</i>

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
44.16	Edge count2 lim	UINT32	32	0...2147483647	-	10000
44.17	Edge count2 div	UINT32	32	0...2147483647	-	1
44.18	Edg cnt2 alm sel	enum	16	0...5	-	Output relay
44.19	Val count1 func	Pb	16	0b00...0b11	-	0b01
44.20	Val count1 src	Val pointer	32	-	-	Speed rpm
44.21	Val count1 lim	UINT32	32	0...2147483647	-	13140000
44.22	Val count1 div	UINT32	32	0...2147483647	-	60
44.23	Val cnt1 alm sel	enum	16	0...1	-	Mot bearing
44.24	Val count2 func	Pb	16	0b00...0b11	-	0b01
44.25	Val count2 src	Val pointer	32	-	-	Speed rpm
44.26	Val count2 lim	UINT32	32	0...2147483647	-	6570000
44.27	Val count2 div	UINT32	32	0...2147483647	-	60
44.28	Val cnt2 alm sel	enum	16	0...1	-	Value2
44.29	Fan ontime lim	UINT32	32	0...35791394.1	h	0.00 h
44.30	Runtime lim	UINT32	32	0...35791394.1	h	0.00 h
44.31	Runtime alm sel	enum	16	1...5	-	Device clean
44.32	kWh inv lim	UINT32	32	0...2147483647	kWh	0 kWh
44.33	kWh inv alm sel	enum	16	1...5	-	Device clean
45 Energy optimising						
45.01	Energy optim	enum	16	0...1	-	Disable
45.02	Energy tariff1	UINT32	32	0...21474836.47	-	0.65
45.06	E tariff unit	enum	16	0...2	-	0
45.07	CO2 Conv factor	REAL	16	0...10	-	0.5
45.08	Reference power	REAL	16	0...1000	%	100.0%
45.09	Energy reset	enum	16	0...1	-	Done
47 Voltage ctrl						
47.01	Overvolt ctrl	enum	16	0...1	-	Enable
47.02	Undervolt ctrl	enum	16	0...1	-	Enable
47.03	SupplyVoltAutold	enum	16	0...1	-	Enable
47.04	Supply voltage	REAL	16	0...1000	V	400.0 V
48 Brake chopper						
48.01	Bc enable	enum	16	0...2	-	Disable
48.02	Bc run-time ena	Bit pointer	32	-	-	C.TRUE
48.03	BrThermTimeConst	REAL24	32	0...10000	s	0 s
48.04	Br power max cnt	REAL24	32	0...10000	kW	0.0000 kW
48.05	R br	REAL24	32	0.1...1000	ohm	0.0000 Ohm
48.06	Br temp faultlim	REAL24	16	0...150	%	105%
48.07	Br temp alarmlim	REAL24	16	0...150	%	95%
49 Data storage						
49.01	Data storage1	UINT32	16	-32768...32768	-	0

276 Additional parameter data

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
49.02	<i>Data storage2</i>	UINT32	16	-32768...32768	-	0
49.03	<i>Data storage3</i>	UINT32	16	-32768...32768	-	0
49.04	<i>Data storage4</i>	UINT32	16	-32768...32768	-	0
49.05	<i>Data storage5</i>	UINT32	32	-2147483647 ... 2147483647	-	0
49.06	<i>Data storage6</i>	UINT32	32	-2147483647 ... 2147483647	-	0
49.07	<i>Data storage7</i>	UINT32	32	-2147483647 ... 2147483647	-	0
49.08	<i>Data storage8</i>	UINT32	32	-2147483647 ... 2147483647	-	0
50 Fieldbus						
50.01	<i>Fba enable</i>	enum	16	0...1	-	<i>Disable</i>
50.02	<i>Comm loss func</i>	enum	16	0...3	-	<i>No</i>
50.03	<i>Comm loss t out</i>	UINT32	16	0.3...6553.5	s	0.3 s
50.04	<i>Fb ref1 modesel</i>	enum	16	0...2	-	<i>Speed</i>
50.05	<i>Fb ref2 modesel</i>	enum	16	0...2	-	<i>Torque</i>
50.06	<i>Fb act1 tr src</i>	Val pointer	32	-	-	P.01.01
50.07	<i>Fb act2 tr src</i>	Val pointer	32	-	-	P.01.06
50.08	<i>Fb sw b12 src</i>	Bit pointer	32	-	-	C.FALSE
50.09	<i>Fb sw b13 src</i>	Bit pointer	32	-	-	C.FALSE
50.10	<i>Fb sw b14 src</i>	Bit pointer	32	-	-	C.FALSE
50.11	<i>Fb sw b15 src</i>	Bit pointer	32	-	-	C.FALSE
50.15	<i>Fb cw used</i>	Val pointer	32	-	-	P.02.22
50.20	<i>Fb main sw func</i>	Pb	16	0b000...0b111	-	0b001
51 FBA settings						
51.01	<i>FBA type</i>	UINT32	16	0...65536	-	0
51.02	<i>FBA par2</i>	UINT32	16	0...65536	-	0
...
51.26	<i>FBA par26</i>	UINT32	16	0...65536	-	0
51.27	<i>FBA par refresh</i>	enum	16	0...1	-	<i>Done</i>
51.28	<i>Par table ver</i>	UINT32	16	0...65536	-	-
51.29	<i>Drive type code</i>	UINT32	16	0...65536	-	-
51.30	<i>Mapping file ver</i>	UINT32	16	0...65536	-	-
51.31	<i>D2FBA comm sta</i>	enum	16	0...6	-	<i>Idle</i>
51.32	<i>FBA comm sw ver</i>	UINT32	16	0...65536	-	-
51.33	<i>FBA appl sw ver</i>	UINT32	16	0...65536	-	-
52 FBA data in						
52.01	<i>FBA data in1</i>	UINT32	16	0...9999	-	0
...
52.12	<i>FBA data in12</i>	UINT32	16	0...9999	-	0

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
53 FBA data out						
53.01	<i>FBA data out1</i>	UINT32	16	0...9999	-	0
...
53.12	<i>FBA data out12</i>	UINT32	16	0...9999	-	0
56 Panel display						
56.01	<i>Signal1 param</i>	UINT32		00.00 ... 255.255	-	01.03
56.02	<i>Signal2 param</i>	UINT32		00.00 ... 255.255	-	01.04
56.03	<i>Signal3 param</i>	UINT32		00.00 ... 255.255	-	01.06
56.04	<i>Signal1 mode</i>	INT32		-1...3	-	<i>Normal</i>
56.05	<i>Signal2 mode</i>	INT32		-1...3	-	<i>Normal</i>
56.06	<i>Signal3 mode</i>	INT32		-1...3	-	<i>Normal</i>
56.07	<i>Local ref unit</i>	UINT32		0...1	-	<i>rpm</i>
57 D2D communication						
57.01	<i>Link mode</i>	enum	16	0...2	-	<i>Disabled</i>
57.02	<i>Comm loss func</i>	enum	16	0...2	-	<i>Alarm</i>
57.03	<i>Node address</i>	UINT32	16	1...62	-	1
57.04	<i>Follower mask 1</i>	UINT32	32	0h00000000 ... 0h7FFFFFFF	-	0h00000000
57.05	<i>Follower mask 2</i>	UINT32	32	0h00000000 ... 0h7FFFFFFF	-	0h00000000
57.06	<i>Ref 1 src</i>	Val pointer	32	-	-	P.03.05
57.07	<i>Ref 2 src</i>	Val pointer	32	-	-	P.03.13
57.08	<i>Follower cw src</i>	Val pointer	32	-	-	P.02.31
57.11	<i>Ref1 msg type</i>	enum	16	0...1	-	<i>Broadcast</i>
57.12	<i>Ref1 mc group</i>	UINT32	16	0...62	-	0
57.13	<i>Next ref1 mc grp</i>	UINT32	16	0...62	-	0
57.14	<i>Nr ref1 mc grps</i>	UINT32	16	1...62	-	1
57.15	<i>D2D com port</i>	enum	16	0...3	-	<i>on-board</i>
58 Embedded Modbus						
58.01	<i>Protocol ena sel</i>	UINT32	32	0...1	-	<i>Modbus RTU</i>
58.03	<i>Node address</i>	UINT32	32	0...247	-	1
58.04	<i>Baud rate</i>	UINT32	32	0...6	-	<i>9600</i>
58.05	<i>Parity</i>	UINT32	32	0...3	-	<i>8 none 1</i>
58.06	<i>Control profile</i>	UINT32	32	0...3	-	<i>ABB Enhanced</i>
58.07	<i>Comm loss t out</i>	UINT32	32	0...60000	ms	600
58.08	<i>Comm loss mode</i>	UINT32	32	0...2	-	<i>None</i>
58.09	<i>Comm loss action</i>	UINT32	32	0...3	-	<i>None</i>
58.10	<i>Refresh settings</i>	UINT32	32	0...1	-	<i>Done</i>
58.11	<i>Reference scale</i>	Pb	16	1...65535	-	100
58.15	<i>Comm diagnostics</i>	Pb	16	0x0000...0xFFFF	-	0x0000
58.16	<i>Received packets</i>	UINT32	32	0...65535	-	0
58.17	<i>Transm packets</i>	UINT32	32	0...65535	-	0

278 Additional parameter data

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
58.18	All packets	UINT16	16	0...65535	-	0
58.19	UART errors	UINT16	16	0...65535	-	0
58.20	CRC errors	UINT16	16	0...65535	-	0
58.21	Raw CW LSW	Pb	16	0x0000...0xFFFF	-	0x0000
58.22	Raw CW MSW	Pb	16	0x0000...0xFFFF	-	0x0000
58.23	Raw SW LSW	Pb	16	0x0000...0xFFFF	-	0x0000
58.24	Raw SW MSW	Pb	16	0x0000...0xFFFF	-	0x0000
58.25	Raw Ref 1 LSW	Pb	16	0x0000...0xFFFF	-	0x0000
58.26	Raw Ref 1 MSW	Pb	16	0x0000...0xFFFF	-	0x0000
58.27	Raw Ref 2 LSW	Pb	16	0x0000...0xFFFF	-	0x0000
58.28	Raw Ref 2 MSW	Pb	16	0x0000...0xFFFF	-	0x0000
58.30	Transmit delay	UINT16	16	0...65535	ms	0
58.31	Ret app errors	UINT16	16	0...1	-	Yes
58.32	Word order	UINT32	32	0...1	-	LSW MSW
58.35	Data I/O 1	UINT16	16	0...9999	-	0
58.36	Data I/O 2	UINT16	16	0...9999	-	0
...
58.58	Data I/O 24	UINT16	16	0...9999	-	0
64 Load analyzer						
64.01	PVL signal	Val pointer	32	-	-	Power inu
64.02	PVL filt time	REAL	16	0...120	s	2.00 s
64.03	Reset loggers	Bit pointer	32	-	-	C.FALSE
64.04	AL signal	Val pointer	32	-	-	Power motor
64.05	AL signal base	REAL	32	0...32768	-	100.00
64.06	PVL peak value1	REAL	32	-32768...32768	-	-
64.07	Date of peak	UINT32	32	01.01.80...	d	-
64.08	Time of peak	UINT32	32	00:00:00...23:59:59	s	-
64.09	Current at peak	REAL	32	-32768...32768	A	-
64.10	Dc volt at peak	REAL	32	0...2000	V	-
64.11	Speed at peak	REAL	32	-32768...32768	rpm	-
64.12	Date of reset	UINT32	32	01.01.80...	d	-
64.13	Time of reset	UINT32	32	00:00:00...23:59:59	s	-
64.14	AL1 0 to 10%	REAL	16	0...100	%	-
64.15	AL1 10 to 20%	REAL	16	0...100	%	-
64.16	AL1 20 to 30%	REAL	16	0...100	%	-
64.17	AL1 30 to 40%	REAL	16	0...100	%	-
64.18	AL1 40 to 50%	REAL	16	0...100	%	-
64.19	AL1 50 to 60%	REAL	16	0...100	%	-
64.20	AL1 60 to 70%	REAL	16	0...100	%	-
64.21	AL1 70 to 80%	REAL	16	0...100	%	-

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
64.22	<i>AL1 80 to 90%</i>	REAL	16	0...100	%	-
64.23	<i>AL1 over 90%</i>	REAL	16	0...100	%	-
64.24	<i>AL2 0 to 10%</i>	REAL	16	0...100	%	-
64.25	<i>AL2 10 to 20%</i>	REAL	16	0...100	%	-
64.26	<i>AL2 20 to 30%</i>	REAL	16	0...100	%	-
64.27	<i>AL2 30 to 40%</i>	REAL	16	0...100	%	-
64.28	<i>AL2 40 to 50%</i>	REAL	16	0...100	%	-
64.29	<i>AL2 50 to 60%</i>	REAL	16	0...100	%	-
64.30	<i>AL2 60 to 70%</i>	REAL	16	0...100	%	-
64.31	<i>AL2 70 to 80%</i>	REAL	16	0...100	%	-
64.32	<i>AL2 80 to 90%</i>	REAL	16	0...100	%	-
64.33	<i>AL2 over 90%</i>	REAL	16	0...100	%	-
90 Enc module sel						
90.01	<i>Encoder 1 sel</i>	enum	16	0...7	-	<i>None</i>
90.02	<i>Encoder 2 sel</i>	enum	16	0...7	-	<i>None</i>
90.04	<i>TTL echo sel</i>	enum	16	0...5	-	<i>Disabled</i>
90.05	<i>Enc cable fault</i>	enum	16	0...2	-	<i>Fault</i>
90.10	<i>Enc par refresh</i>	enum	16	0...1	-	<i>Done</i>
91 Absol enc conf						
91.01	<i>Sine cosine nr</i>	UINT32	16	0...65535	-	0
91.02	<i>Abs enc interf</i>	enum	16	0...5	-	<i>None</i>
91.03	<i>Rev count bits</i>	UINT32	16	0...32	-	0
91.04	<i>Pos data bits</i>	UINT32	16	0...32	-	0
91.05	<i>Refmark ena</i>	enum	16	0...1	-	<i>False</i>
91.10	<i>Hiperface parity</i>	enum	16	0...1	-	<i>Odd</i>
91.11	<i>Hiperf baudrate</i>	enum	16	0...3	-	<i>9600</i>
91.12	<i>Hiperf node addr</i>	UINT32	16	0...255	-	64
91.20	<i>SSI clock cycles</i>	UINT32	16	2...127	-	2
91.21	<i>SSI position msb</i>	UINT32	16	1...126	-	1
91.22	<i>SSI revol msb</i>	UINT32	16	1...126	-	1
91.23	<i>SSI data format</i>	enum	16	0...1	-	<i>binary</i>
91.24	<i>SSI baud rate</i>	enum	16	0...5	-	<i>100 kbit/s</i>
91.25	<i>SSI mode</i>	enum	16	0...1	-	<i>Initial pos.</i>
91.26	<i>SSI transmit cyc</i>	enum	16	0...5	-	<i>100 µs</i>
91.27	<i>SSI zero phase</i>	enum	16	0...3	-	<i>315-45 deg</i>
91.30	<i>Endat mode</i>	enum	16	0...1	-	<i>Initial pos.</i>
91.31	<i>Endat max calc</i>	enum	16	0...3	-	<i>50 ms</i>
92 Resolver conf						
92.01	<i>Resolv polepairs</i>	UINT32	16	1...32	-	1
92.02	<i>Exc signal ampl</i>	UINT32	16	4...12	Vrms	4.0 Vrms

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
92.03	<i>Exc signal freq</i>	UINT32	16	1...20	kHz	1 kHz
93 Pulse enc conf						
93.01	<i>Enc1 pulse nr</i>	UINT32	16	0...65535	-	0
93.02	<i>Enc1 type</i>	enum	16	0...1	-	<i>Quadrature</i>
93.03	<i>Enc1 sp CalcMode</i>	enum	16	0...5	-	<i>Auto rising</i>
93.11	<i>Enc2 pulse nr</i>	UINT32	16	0...65535	-	0
93.12	<i>Enc2 type</i>	enum	16	0...1	-	<i>Quadrature</i>
93.13	<i>Enc2 sp CalcMode</i>	enum	16	0...5	-	<i>Auto rising</i>
94 Ext IO conf						
94.01	<i>Ext IO1 sel</i>	UINT32	16	0...3	-	<i>None</i>
94.02	<i>Ext IO2 sel</i>	UINT32	16	0...3	-	<i>None</i>
95 Hw configuration						
95.01	<i>Ctrl boardSupply</i>	enum	16	0...1	-	<i>Internal 24V</i>
95.03	<i>Temp inu ambient</i>	INT32	16	0...55	°C	40 °C
97 User motor par						
97.01	<i>Use given params</i>	enum	16	0...3	-	<i>NoUserPars</i>
97.02	<i>Rs user</i>	REAL24	32	0...0.5	p.u.	0.00000 p.u.
97.03	<i>Rr user</i>	REAL24	32	0...0.5	p.u.	0.00000 p.u.
97.04	<i>Lm user</i>	REAL24	32	0...10	p.u.	0.00000 p.u.
97.05	<i>SigmaL user</i>	REAL24	32	0...1	p.u.	0.00000 p.u.
97.06	<i>Ld user</i>	REAL24	32	0...10	p.u.	0.00000 p.u.
97.07	<i>Lq user</i>	REAL24	32	0...10	p.u.	0.00000 p.u.
97.08	<i>Pm flux user</i>	REAL24	32	0...2	p.u.	0.00000 p.u.
97.09	<i>Rs user SI</i>	REAL24	32	0...100	ohm	0.00000 Ohm
97.10	<i>Rr user SI</i>	REAL24	32	0...100	ohm	0.00000 Ohm
97.11	<i>Lm user SI</i>	REAL24	32	0...100000	mH	0.00 mH
97.12	<i>SigL user SI</i>	REAL24	32	0...100000	mH	0.00 mH
97.13	<i>Ld user SI</i>	REAL24	32	0...100000	mH	0.00 mH
97.14	<i>Lq user SI</i>	REAL24	32	0...100000	mH	0.00 mH
97.20	<i>PM angle offset</i>	REAL	32	0...360	° (el.)	0°
99 Start-up data						
99.01	<i>Language</i>	enum	16	-	-	<i>English</i>
99.04	<i>Motor type</i>	enum	16	0...1	-	<i>AM</i>
99.05	<i>Motor ctrl mode</i>	enum	16	0...1	-	<i>DTC</i>
99.06	<i>Mot nom current</i>	REAL	32	0...6400	A	0.0 A
99.07	<i>Mot nom voltage</i>	REAL	32	$1/6 \dots 2 \times U_N$	V	0.0 V
99.08	<i>Mot nom freq</i>	REAL	32	5...500	Hz	0.0 Hz
99.09	<i>Mot nom speed</i>	REAL	32	0...10000	rpm	0 rpm
99.10	<i>Mot nom power</i>	REAL	32	0...10000	kW or hp	0.00 kW
99.11	<i>Mot nom cosfii</i>	REAL24	32	0...1	-	0.00

No.	Name	Type	Data len.	Range	Unit	Default (Factory macro)
99.12	<i>Mot nom torque</i>	INT32	32	0...2147483.647	Nm	0.000 Nm
99.13	<i>IDrun mode</i>	enum	16	0...5	-	<i>No</i>



Fault tracing efesotomasyon.com

What this chapter contains

The chapter lists the alarm (warning) and fault messages including possible causes and corrective actions.

The alarm/fault code is displayed on the control panel of the drive, as well as the DriveStudio PC tool. An alarm or a fault message indicates abnormal drive status. Most alarm and fault causes can be identified and corrected using the information in this chapter. If not, an ABB representative should be contacted.

In this chapter, the alarms and faults are sorted by the four-digit code. The hexadecimal code in brackets that follows the alarm/fault message is for fieldbus communication.

Safety



WARNING! Only qualified electricians are allowed to maintain the drive. The *Safety Instructions* on the first pages of the appropriate *Hardware Manual* must be read before you start working with the drive.

How to reset

The drive can be reset either by pressing the RESET key on the control panel or PC tool, or by switching the supply voltage off for a while. When the fault has been removed, the motor can be restarted.

A fault can also be reset from an external source selected by parameter [10.10 Fault reset sel.](#)

Fault history

When a fault is detected, it is stored in the fault logger with a time stamp. The fault history stores information on the 16 latest faults of the drive. Three of the latest faults are stored at the beginning of a power switch off.

Parameters [08.01 Active fault](#) and [08.02 Last fault](#) store the fault codes of the most recent faults.

Alarms can be monitored via alarm words [08.05 Alarm word1](#) ... [08.08 Alarm word4](#). Alarm information is lost at power switch off or fault reset.

Alarm messages generated by the drive

Code	Alarm (fieldbus code)	Cause	What to do
2000	BRAKE START TORQUE (0x7185) Programmable fault: 42.12 Brake fault func	Mechanical brake alarm. Alarm is activated if required motor starting torque (42.08 Brake open torq) is not achieved.	Check brake open torque setting, parameter 42.08 . Check drive torque and current limits. See parameter group 20 Limits .
2001	BRAKE NOT CLOSED (0x7186) Programmable fault: 42.12 Brake fault func	Mechanical brake control alarm. Alarm is activated e.g. if brake acknowledgement is not as expected during brake closing.	Check mechanical brake connection. Check mechanical brake settings in parameter group 42 Mech brake ctrl . To determine whether problem is with acknowledgement signal or brake, check if brake is closed or open.
2002	BRAKE NOT OPEN (0x7187) Programmable fault: 42.12 Brake fault func	Mechanical brake control alarm. Alarm is activated e.g. if brake acknowledgement is not as expected during brake opening.	Check mechanical brake connection. Check mechanical brake settings in parameter group 42 Mech brake ctrl . To determine whether problem is with acknowledgement signal or brake, check if brake is closed or open.
2003	SAFE TORQUE OFF (0xFF7A) Programmable fault: 30.07 Sto diagnostic	Safe torque off function is active, i.e. safety circuit signal(s) connected to connector XSTO is lost.	Check safety circuit connections. For more information, see the appropriate drive <i>Hardware Manual</i> , description of parameter 30.07 (page 190), and <i>Application guide - Safe torque off function for ACSM1, ACS850 and ACQ810 drives</i> (3AFE68929814 [English]).
2004	STO MODE CHANGE (0xFF7A)	Error in changing Safe torque off supervision, i.e. parameter 30.07 Sto diagnostic setting could not be changed to value Alarm .	Contact your local ABB representative.

Code	Alarm (fieldbus code)	Cause	What to do
2005	MOTOR TEMPERATURE (0x4310) Programmable fault: 31.01 Mot temp1 prot	Estimated motor temperature (based on motor thermal model) has exceeded alarm limit defined by parameter 31.03 Mot temp1 almLim .	Check motor ratings and load. Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc. Check value of alarm limit. Check motor thermal model settings (parameters 31.09...31.14).
		Measured motor temperature has exceeded alarm limit defined by parameter 31.03 Mot temp1 almLim .	Check that actual number of sensors corresponds to value set by parameter 31.02 Mot temp1 src . Check motor ratings and load. Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc. Check value of alarm limit.
2006	EMERGENCY OFF (0xF083)	Drive has received emergency OFF2 command.	To restart drive, activate run enable signal (source selected by parameter 10.11 Run enable) and start drive.
2007	RUN ENABLE (0xFF54)	No run enable signal is received.	Check setting of parameter 10.11 Run enable . Switch signal on (e.g. in the fieldbus Control Word) or check wiring of selected source.
2008	ID-RUN (0xFF84)	Motor identification run is on.	This alarm belongs to normal start-up procedure. Wait until drive indicates that motor identification is completed.
		Motor identification is required.	This alarm belongs to normal start-up procedure. Select how motor identification should be performed, parameter 99.13 IDrun mode . Start identification routines by pressing Start key.
2009	EMERGENCY STOP (0xF081)	Drive has received emergency stop command (OFF1/OFF3).	Check that it is safe to continue operation. Return emergency stop push button to normal position (or adjust the fieldbus Control Word accordingly). Restart drive.
2011	BR OVERHEAT (0x7112)	Brake resistor temperature has exceeded alarm limit defined by parameter 48.07 Br temp alarmlim .	Stop drive. Let resistor cool down. Check resistor overload protection function settings (parameters 48.01...48.05). Check alarm limit setting, parameter 48.07 Br temp alarmlim . Check that braking cycle meets allowed limits.

Code	Alarm (fieldbus code)	Cause	What to do
2012	BC OVERHEAT (0x7181)	Brake chopper IGBT temperature has exceeded internal alarm limit.	Let chopper cool down. Check for excessive ambient temperature. Check for cooling fan failure. Check for obstructions in the air flow. Check the dimensioning and cooling of the cabinet. Check resistor overload protection function settings (parameters 48.01...48.05). Check that braking cycle meets allowed limits. Check that drive supply AC voltage is not excessive.
2013	DEVICE OVERTEMP (0x4210)	Measured drive temperature has exceeded internal alarm limit.	Check ambient conditions. Check air flow and fan operation. Check heatsink fins for dust pick-up. Check motor power against unit power.
2014	INTBOARD OVERTEMP (0x7182)	Interface board (between power unit and control unit) temperature has exceeded internal alarm limit.	Let drive cool down. Check for excessive ambient temperature. Check for cooling fan failure. Check for obstructions in the air flow. Check the dimensioning and cooling of the cabinet.
2015	BC MOD OVERTEMP (0x7183)	Input bridge or brake chopper temperature has exceeded internal alarm limit.	Let drive cool down. Check for excessive ambient temperature. Check for cooling fan failure. Check for obstructions in the air flow. Check the dimensioning and cooling of the cabinet.
2017	FIELDBUS COMM (0x7510) Programmable fault: 50.02 Comm loss func	Cyclical communication between drive and fieldbus adapter module or between PLC and fieldbus adapter module is lost.	Check status of fieldbus communication. See the appropriate <i>User's Manual</i> of fieldbus adapter module. Check settings of parameter group 50 Fieldbus . Check cable connections. Check if communication master is able to communicate.
2018	LOCAL CTRL LOSS (0x5300) Programmable fault: 30.03 Local ctrl loss	Control panel or PC tool selected as active control location for drive has ceased communicating.	Check PC tool or control panel connection. Check control panel connector. Replace control panel in mounting platform.
2019	AI SUPERVISION (0x8110) Programmable fault: 13.32 Ai superv func	An analogue input has reached limit defined by parameter 13.33 Ai superv cw .	Check analogue input source and connections. Check analogue input minimum and maximum limit settings.
2020	FB PAR CONF (0x6320)	The drive does not have a functionality requested by PLC, or requested functionality has not been activated.	Check PLC programming. Check settings of parameter group 50 Fieldbus .

Code	Alarm (fieldbus code)	Cause	What to do
2021	NO MOTOR DATA (0x6381)	Parameters in group 99 have not been set.	Check that all the required parameters in group 99 have been set. Note: It is normal for this alarm to appear during the start-up until the motor data is entered.
2022	ENCODER 1 FAILURE (0x7301)	Encoder 1 has been activated by parameter but the encoder interface (FEN-xx) cannot be found.	Check parameter <i>90.01 Encoder 1 sel</i> setting corresponds to actual encoder interface 1 (FEN-xx) installed in drive Slot 1/2 (parameter <i>09.20 Option slot1</i> / <i>09.21 Option slot2</i>). Note: The new setting will only take effect after parameter <i>90.10 Enc par refresh</i> is used or after the JCU Control Unit is powered up the next time.
2023	ENCODER 2 FAILURE (0x7381)	Encoder 2 has been activated by parameter but the encoder interface (FEN-xx) cannot be found.	Check parameter <i>90.02 Encoder 2 sel</i> setting corresponds to actual encoder interface 1 (FEN-xx) installed in drive Slot 1/2 (parameter <i>09.20 Option slot1</i> / <i>09.21 Option slot2</i>). Note: The new setting will only take effect after parameter <i>90.10 Enc par refresh</i> is used or after the JCU Control Unit is powered up the next time.
2027	FEN TEMP MEAS FAILURE (0x7385)	Error in temperature measurement when temperature sensor (KTY or PTC) connected to encoder interface FEN-xx is used.	Check that parameter <i>31.02 Mot temp1 src</i> / <i>31.06 Mot temp2 src</i> setting corresponds to actual encoder interface installation (<i>09.20 Option slot1</i> / <i>09.21 Option slot2</i>): If one FEN-xx module is used: - Parameter <i>31.02 Mot temp1 src</i> / <i>31.06 Mot temp2 src</i> must be set either to <i>KTY 1st FEN</i> or <i>PTC 1st FEN</i> . The FEN-xx module can be in either Slot 1 or Slot 2. If two FEN-xx modules are used: - When parameter <i>31.02 Mot temp1 src</i> / <i>31.06 Mot temp2 src</i> is set to <i>KTY 1st FEN</i> or <i>PTC 1st FEN</i> , the encoder installed in drive Slot 1 is used. - When parameter <i>31.02 Mot temp1 src</i> / <i>31.06 Mot temp2 src</i> is set to <i>KTY 2nd FEN</i> or <i>PTC 2nd FEN</i> , the encoder installed in drive Slot 2 is used.
		Error in temperature measurement when KTY sensor connected to encoder interface FEN-01 is used.	FEN-01 does not support temperature measurement with KTY sensor. Use PTC sensor or other encoder interface module.

Code	Alarm (fieldbus code)	Cause	What to do
2030	RESOLVER AUTOTUNE ERR (0x7388)	Resolver autotuning routines, which are automatically started when resolver input is activated for the first time, have failed.	Check cable between resolver and resolver interface module (FEN-21) and order of connector signal wires at both ends of cable. Check resolver parameter settings. For resolver parameters and information, see parameter group 92 Resolver conf . Note: Resolver autotuning routines should always be performed after resolver cable connection has been modified. Autotuning routines can be activated by setting parameter 92.02 Exc signal ampl or 92.03 Exc signal freq , and then setting parameter 90.10 Enc par refresh to <i>Configure</i> .
2031	ENCODER 1 CABLE (0x7389)	Encoder 1 cable fault detected.	Check cable between FEN-xx interface and encoder 1. After any modifications in cabling, re-configure interface by switching drive power off and on, or by activating parameter 90.10 Enc par refresh .
2032	ENCODER 2 CABLE (0x738A)	Encoder 2 cable fault detected.	Check cable between FEN-xx interface and encoder 2. After any modifications in cabling, re-configure interface by switching drive power off and on, or by activating parameter 90.10 Enc par refresh .
2033	D2D COMMUNICATION (0x7520) Programmable fault: 57.02 Comm loss func	On the master drive: The drive has not been replied to by an activated follower for five consecutive polling cycles.	Check that all drives that are polled (parameters 57.04 Follower mask 1 and 57.05 Follower mask 2) on the drive-to-drive link are powered, properly connected to the link, and have the correct node address. Check the drive-to-drive link wiring.
		On a follower drive: The drive has not received new reference 1 and/or 2 for five consecutive reference handling cycles.	Check the settings of parameters 57.06 Ref 1 src and 57.07 Ref 2 src on the master drive. Check the drive-to-drive link wiring.
2034	D2D BUFFER OVERLOAD (0x7520) Programmable fault: 57.02 Comm loss func	Transmission of drive-to-drive references failed because of message buffer overflow.	Contact your local ABB representative.
2035	PS COMM (0x5480)	Communication errors detected between the JCU Control Unit and the power unit of the drive.	Check the connections between the JCU Control Unit and the power unit.
2036	RESTORE (0x6300)	Restoration of backed-up parameters failed.	Contact your local ABB representative.
2037	CUR MEAS CALIBRATION (0x2280)	Current measurement calibration will occur at next start.	Informative alarm.
2038	AUTOPHASING (0x3187)	Autophasing will occur at next start.	Informative alarm.

Code	Alarm (fieldbus code)	Cause	What to do
2039	EARTH FAULT (0x2330) Programmable fault: 30.05 Earth fault	Drive has detected load unbalance typically due to earth fault in motor or motor cable.	Check there are no power factor correction capacitors or surge absorbers in motor cable. Check for an earth fault in motor or motor cables by measuring the insulation resistances of motor and motor cable. If no earth fault can be detected, contact your local ABB representative.
2040	AUTORESET (0x6080)	A fault is to be autoreset.	Informative alarm. See parameter group 32 Automatic reset .
2041	MOTOR NOM VALUE (0x6383)	The motor configuration parameters are set incorrectly.	Check the settings of the motor configuration parameters in group 99.
		The drive is not dimensioned correctly.	Check that the drive is sized correctly for the motor.
2042	D2D CONFIG (0x7583)	The settings of drive-to-drive link configuration parameters (group 57) are incompatible.	Check the settings of the parameters in group 57 D2D communication .
2043	STALL (0x7121) Programmable fault: 30.09 Stall function	Motor is operating in stall region because of e.g. excessive load or insufficient motor power.	Check motor load and drive ratings. Check fault function parameters.
2044	LCURVE (0x2312) Programmable fault: 34.01 Overload func / 34.02 Underload func	Overload or underload limit has been exceeded.	Check the settings of the parameters in group 34 User load curve .
2045	LCURVE PAR (0x6320)	The load curve has been incorrectly or inconsistently defined.	Check the settings of the parameters in group 34 User load curve .
2046	FLUX REF PAR (0x6320)	The U/f (voltage/frequency) curve has been incorrectly or inconsistently defined.	Check the settings of the parameters in group 38 Flux ref .
2047	SPEED FEEDBACK (0x8480)	No speed feedback is received.	Check the settings of the parameters in group 19 Speed calculation . Check encoder installation. See the description of fault 0039 for more information.
2048	OPTION COMM LOSS (0x7000)	Communication between drive and option module (FEN-xx and/or FIO-xx) is lost.	Check that option modules are properly connected to Slot 1 and (or) Slot 2. Check that option modules or Slot 1/2 connectors are not damaged. To determine whether module or connector is damaged: Test each module individually in Slot 1 and Slot 2.

Code	Alarm (fieldbus code)	Cause	What to do
2049	MOTTEMPAL2 (0x4313) Programmable fault: 31.05 Mot temp2 prot	Estimated motor temperature (based on motor thermal model) has exceeded alarm limit defined by parameter 31.07 Mot temp2 almLim .	Check motor ratings and load. Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc. Check value of alarm limit. Check motor thermal model settings (parameters 31.09...31.14).
		Measured motor temperature has exceeded alarm limit defined by parameter 31.07 Mot temp2 almLim .	Check that actual number of sensors corresponds to value set by parameter 31.06 Mot temp2 src . Check motor ratings and load. Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc. Check value of alarm limit.
2050	IGBTOLALARM (0x5482)	Excessive IGBT junction to case temperature. This alarm protects the IGBT(s) and can be activated by a short circuit in the motor cable.	Check motor cable.
2051	IGBTTEMPALARM (0x4210)	Drive IGBT temperature is excessive.	Check ambient conditions. Check air flow and fan operation. Check heatsink fins for dust pick-up. Check motor power against drive power.
2052	COOLALARM (0x4290)	Drive module temperature is excessive.	Check ambient temperature. If it exceeds 40 °C (104 °F), ensure that load current does not exceed derated load capacity of drive. See the appropriate <i>Hardware Manual</i> . Check drive module cooling air flow and fan operation. Check inside of cabinet and heatsink of drive module for dust pick-up. Clean whenever necessary.
2053	MENU CHG PASSWORD REQ (0x6F81)	Loading a parameter listing requires a password.	Enter password at parameter 16.03 Pass code .
2054	MENU CHANGED (0x6F82)	A different parameter listing is being loaded.	Informative alarm.
2055	DEVICE CLEAN (0x5080)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2056	COOLING FAN (0x5081)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2057	ADD COOLING (0x5082)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2058	CABINET FAN (0x5083)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2059	DC CAPACITOR (0x5084)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2060	MOTOR BEARING (0x738C)	Maintenance counter alarm.	See parameter group 44 Maintenance .

Code	Alarm (fieldbus code)	Cause	What to do
2061	MAIN CONTACTOR (0x548D)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2062	RELAY OUTPUT SW (0x548E)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2063	MOTOR START COUNT (0x6180)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2064	POWER UP COUNT (0x6181)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2065	DC CHARGE COUNT (0x6182)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2066	ONTIME1 ALARM (0x5280)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2067	ONTIME2 ALARM (0x5281)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2068	EDGE1 ALARM (0x5282)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2069	EDGE2 ALARM (0x5283)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2070	VALUE1 ALARM (0x5284)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2071	VALUE2 ALARM (0x5285)	Maintenance counter alarm.	See parameter group 44 Maintenance .
2072	DC NOT CHARGED (0x3250)	The voltage of the intermediate DC circuit has not yet risen to operating level.	Wait for the DC voltage to rise.
2073	AUTOTUNE FAILED (0x8481)	Speed controller autotune routine did not finish successfully.	See parameter 23.20 PI tune mode .
2074	START INTERLOCK (0xF082)	No Start interlock signal received.	Check circuit connected to DIIL input.
2076	TEMP MEAS FAILURE (0x4211)	Problem with internal temperature measurement of the drive.	Contact your local ABB representative.
2077	EFB COMM LOSS ALARM	Embedded fieldbus interface has been taken into use, and there is a communication break between the drive and the master station.	Check: <ul style="list-style-type: none"> • selection of the parameter which enables/disables EFB communication (58.01 Protocol enable) • EFB connection at terminal XD2D on the JCON board • status of the fieldbus master (online/offline) • settings of the communication supervision function (parameter 58.09 Comm loss action).
2078	TEMP DIFFERENCE (0x4212)	High temperature difference between the IGBTs of different phases.	Check cooling and the fan.

Fault messages generated by the drive

Code	Fault (fieldbus code)	Cause	What to do
0001	OVERCURRENT (0x2310)	Output current has exceeded internal fault limit.	<p>Check motor load.</p> <p>Check acceleration times in parameter group 22 Speed ref ramp.</p> <p>Check motor and motor cable (including phasing and delta/star connection).</p> <p>Check that the start-up data in parameter group 99 corresponds to the motor rating plate.</p> <p>Check that there are no power factor correction capacitors or surge absorbers in motor cable.</p> <p>Check encoder cable (including phasing).</p>
0002	DC OVERVOLTAGE (0x3210)	Excessive intermediate circuit DC voltage	<p>Check that overvoltage controller is on, parameter 47.01 Overvolt ctrl.</p> <p>Check that the supply (input power) voltage matches the drive nominal input voltage.</p> <p>Check mains for static or transient overvoltage.</p> <p>Check brake chopper and resistor (if used).</p> <p>Check deceleration time.</p> <p>Use coast-to-stop function (if applicable).</p> <p>Retrofit frequency converter with brake chopper and brake resistor.</p>
0004	SHORT CIRCUIT (0x2340)	Short-circuit in motor cable(s) or motor	<p>Check motor and motor cable.</p> <p>Check there are no power factor correction capacitors or surge absorbers in motor cable.</p>
0005	DC UNDERVOLTAGE (0x3220)	Intermediate circuit DC voltage is not sufficient due to missing mains phase, blown fuse or rectifier bridge internal fault.	<p>Check mains supply and fuses.</p>
0006	EARTH FAULT (0x2330) Programmable fault: 30.05 Earth fault	Drive has detected load unbalance typically due to earth fault in motor or motor cable.	<p>Check there are no power factor correction capacitors or surge absorbers in motor cable.</p> <p>Check that there is no earth fault in motor or motor cables:</p> <ul style="list-style-type: none"> - measure insulation resistances of motor and motor cable. <p>If no earth fault can be detected, contact your local ABB representative.</p>
0007	FAN FAULT (0xFF83)	Fan is not able to rotate freely or fan is disconnected. Fan operation is monitored by measuring fan current.	<p>Check fan operation and connection.</p>
0009	BC WIRING (0x7111)	Brake resistor short circuit or brake chopper control fault	<p>Check brake chopper and brake resistor connection.</p> <p>Ensure brake resistor is not damaged.</p>
0010	BC SHORT CIRCUIT (0x7113)	Short circuit in brake chopper IGBT	<p>Replace brake chopper.</p> <p>Ensure brake resistor is connected and not damaged.</p>

Code	Fault (fieldbus code)	Cause	What to do
0011	BC OVERHEAT (0x7181)	Brake chopper IGBT temperature has exceeded internal fault limit.	Let chopper cool down. Check for excessive ambient temperature. Check for cooling fan failure. Check for obstructions in the air flow. Check the dimensioning and cooling of the cabinet. Check resistor overload protection function settings (parameters 48.01...48.05). Check that braking cycle meets allowed limits. Check that drive supply AC voltage is not excessive.
0012	BR OVERHEAT (0x7112)	Brake resistor temperature has exceeded fault limit defined by parameter 48.06 Br temp faultlim .	Stop drive. Let resistor cool down. Check resistor overload protection function settings (parameters 48.01...48.05). Check fault limit setting, parameter 48.06 Br temp faultlim . Check that braking cycle meets allowed limits.
0013	CURR MEAS GAIN (0x3183)	Difference between output phase U2 and W2 current measurement gain is too great.	Contact your local ABB representative.
0014	CABLE CROSS CON (0x3181) Programmable fault: 30.08 Cross connection	Incorrect input power and motor cable connection (i.e. input power cable is connected to drive motor connection).	Check input power connections.
0015	SUPPLY PHASE (0x3130) Programmable fault: 30.06 Suppl phs loss	Intermediate circuit DC voltage is oscillating due to missing input power line phase or blown fuse.	Check input power line fuses. Check for input power supply imbalance.
0016	MOTOR PHASE (0x3182) Programmable fault: 30.04 Mot phase loss	Motor circuit fault due to missing motor connection (all three phases are not connected).	Connect motor cable.
0017	ID-RUN FAULT (0xFF84)	Motor ID run is not completed successfully.	Check the fault logger for a fault code extension. See appropriate actions for each extension below.
	Extension: 1	The ID run cannot be completed because the maximum current setting and/or internal current limit of the drive is too low.	Check setting of parameters 99.06 Mot nom current and 20.05 Maximum current . Make sure that 20.05 Maximum current > 99.06 Mot nom current . Check that the drive is dimensioned correctly according to the motor.

Code	Fault (fieldbus code)	Cause	What to do
	Extension: 2	The ID run cannot be completed because the maximum speed setting and/or calculated field weakening point is too low.	Check setting of parameters 99.07 Mot nom voltage , 99.08 Mot nom freq , 99.09 Mot nom speed , 20.01 Maximum speed and 20.02 Minimum speed . Make sure that <ul style="list-style-type: none"> • 20.01 Maximum speed > (0.55 × 99.09 Mot nom speed) > (0.50 × synchronous speed), • 20.02 Minimum speed ≤ 0, and • supply voltage ≥ (0.66 × 99.07 Mot nom voltage).
	Extension: 3	The ID run cannot be completed because the maximum torque setting is too low.	Check setting of parameter 99.12 Mot nom torque and torque limits defined in parameter group 20 Limits . Make sure that the active maximum torque (selected by 20.06 Torq lim sel) > 100%.
	Extension: 4	Current measurement calibration did not finish within reasonable time.	Contact your local ABB representative.
	Extension: 5...8	Internal error.	Contact your local ABB representative.
	Extension: 9	Asynchronous motors only: Acceleration did not finish within reasonable time.	Contact your local ABB representative.
	Extension: 10	Asynchronous motors only: Deceleration did not finish within reasonable time.	Contact your local ABB representative.
	Extension: 11	Asynchronous motors only: Speed dropped to zero during ID run.	Contact your local ABB representative.
	Extension: 12	Permanent magnet motors only: First acceleration did not finish within reasonable time.	Contact your local ABB representative.
	Extension: 13	Permanent magnet motors only: Second acceleration did not finish within reasonable time.	Contact your local ABB representative.
	Extension: 14...16	Internal error.	Contact your local ABB representative.
0018	CURR U2 MEAS (0x3184)	Measured offset error of U2 output phase current measurement is too great. (Offset value is updated during current calibration.)	Contact your local ABB representative.
0019	CURR V2 MEAS (0x3185)	Measured offset error of V2 output phase current measurement is too great. (Offset value is updated during current calibration.)	Contact your local ABB representative.
0020	CURR W2 MEAS (0x3186)	Measured offset error of W2 output phase current measurement is too great. (Offset value is updated during current calibration.)	Contact your local ABB representative.

Code	Fault (fieldbus code)	Cause	What to do
0021	STO1 LOST (0x8182)	Safe torque off function is active, i.e. safety circuit signal 1 connected between XSTO:1 and XSTO:3 is lost.	Check safety circuit connections. For more information, see the appropriate drive <i>Hardware Manual</i> , description of parameter 30.07 (page 190), and <i>Application guide - Safe torque off function for ACSM1, ACS850 and ACQ810 drives</i> (3AFE68929814 [English]).
0022	STO2 LOST (0x8183)	Safe torque off function is active, i.e. safety circuit signal 2 connected between XSTO:2 and XSTO:4 is lost.	
0023	STO MODE CHANGE (0xFF7A)	Error in changing Safe torque off supervision, i.e. parameter 30.07 Sto diagnostic setting could not be changed to value <i>Fault</i> .	Contact your local ABB representative.
0024	INTBOARD OVERTEMP (0x7182)	Interface board (between power unit and control unit) temperature has exceeded internal fault limit.	Let drive cool down. Check for excessive ambient temperature. Check for cooling fan failure. Check for obstructions in the air flow. Check the dimensioning and cooling of the cabinet.
0025	BC MOD OVERTEMP (0x7183)	Input bridge or brake chopper temperature has exceeded internal fault limit.	Let drive cool down. Check for excessive ambient temperature. Check for cooling fan failure. Check for obstructions in the air flow. Check the dimensioning and cooling of the cabinet.
0026	AUTOPHASING (0x3187)	Autophasing routine (see section Autophasing on page 67) failed.	Try other autophasing modes (see parameter 11.07 Autophasing mode) if possible.
0027	PU LOST (0x5400)	Connection between the JCU Control Unit and the power unit of the drive is lost.	Check the connections between the JCU Control Unit and the power unit.
0028	PS COMM (0x5480)	Communication errors detected between the JCU Control Unit and the power unit of the drive.	Check the connections between the JCU Control Unit and the power unit.
0030	EXTERNAL (0x9000)	Fault in external device. (This information is configured through one of programmable digital inputs.)	Check external devices for faults. Check parameter 30.01 External fault setting.
0031	SAFE TORQUE OFF (0xFF7A) Programmable fault: 30.07 Sto diagnostic	Safe torque off function is active, i.e. safety circuit signal(s) connected to connector XSTO is lost during start or run, or while drive is stopped and parameter 30.07 Sto diagnostic is set to <i>Fault</i> .	Check safety circuit connections. For more information, see the appropriate drive <i>Hardware Manual</i> , and <i>Application guide - Safe torque off function for ACSM1, ACS850 and ACQ810 drives</i> (3AFE68929814 [English]).

Code	Fault (fieldbus code)	Cause	What to do
0032	OVERSPEED (0x7310)	Motor is turning faster than highest allowed speed due to incorrectly set minimum/maximum speed, insufficient braking torque or changes in load when using torque reference.	Check minimum/maximum speed settings, parameters 20.01 Maximum speed and 20.02 Minimum speed . Check adequacy of motor braking torque. Check applicability of torque control. Check need for brake chopper and resistor(s).
0033	BRAKE START TORQUE (0x7185) Programmable fault: 42.12 Brake fault func	Mechanical brake fault. Fault is activated if required motor starting torque (42.08 Brake open torq) is not achieved.	Check brake open torque setting, parameter 42.08 . Check drive torque and current limits. See parameter group 20 Limits .
0034	BRAKE NOT CLOSED (0x7186) Programmable fault: 42.12 Brake fault func	Mechanical brake control fault. Activated e.g. if brake acknowledgement is not as expected during brake closing.	Check mechanical brake connection. Check mechanical brake settings in parameter group 42 Mech brake ctrl . To determine whether problem is with acknowledgement signal or brake, check if brake is closed or open.
0035	BRAKE NOT OPEN (0x7187) Programmable fault: 42.12 Brake fault func	Mechanical brake control fault. Activated e.g. if brake acknowledgement is not as expected during brake opening.	Check mechanical brake connection. Check mechanical brake settings in parameter group 42 Mech brake ctrl . To determine whether problem is with acknowledgement signal or brake, check if brake is closed or open.
0036	LOCAL CTRL LOSS (0x5300) Programmable fault: 30.03 Local ctrl loss	Control panel or PC tool selected as active control location for drive has ceased communicating.	Check PC tool or control panel connection. Check control panel connector. Replace control panel in mounting platform.
0037	NVMEM CORRUPTED (0x6320)	Drive internal fault. Note: This fault cannot be reset.	Check the fault logger for a fault code extension. See appropriate actions for each extension below. *Refer to Application programming for ACS850 drives (3AUA0000078664 [English]).
	Extension: 2051	Total number of parameters (including unused space between parameters) exceeds firmware maximum.	*Move parameters from the firmware groups to the application groups. *Reduce the number of parameters.
	Extension: Other	Drive internal fault.	Contact your local ABB representative.
0038	OPTIONCOMM LOSS (0x7000)	Communication between drive and option module (FEN-xx and/or FIO-xx) is lost.	Check that option modules are properly connected to Slot 1 and (or) Slot 2. Check that option modules or Slot 1/2 connectors are not damaged. To determine whether module or connector is damaged: Test each module individually in Slot 1 and Slot 2.

Code	Fault (fieldbus code)	Cause	What to do
0039	ENCODER 1 (0x7301)	Encoder 1 feedback fault.	<p>If fault appears during first start-up before encoder feedback is used:</p> <ul style="list-style-type: none"> - Check cable between encoder and encoder interface module (FEN-xx) and order of connector signal wires at both ends of cable. <p>If fault appears after encoder feedback has already been used or during drive run:</p> <ul style="list-style-type: none"> - Check that encoder connection wiring or encoder is not damaged. - Check that encoder interface module (FEN-xx) connection or module is not damaged. - Check earthings (when disturbances are detected in communication between encoder interface module and encoder). <p>For more information on encoders, see parameter groups 90 Enc module sel, 92 Resolver conf and 93 Pulse enc conf.</p>
0040	ENCODER 2 (0x7381)	Encoder 2 feedback fault.	See fault 0039 .
0045	FIELDBUS COMM (0x7510) Programmable fault: 50.02 Comm loss func	Cyclical communication between drive and fieldbus adapter module or between PLC and fieldbus adapter module is lost.	<p>Check status of fieldbus communication. See the appropriate <i>User's Manual</i> of fieldbus adapter module.</p> <p>Check settings of parameter group 50 Fieldbus.</p> <p>Check cable connections.</p> <p>Check if communication master is able to communicate.</p>
0046	FB MAPPING FILE (0x6306)	Drive internal fault	Contact your local ABB representative.
0047	MOTOR OVERTEMP (0x4310) Programmable fault: 31.01 Mot temp1 prot	Estimated motor temperature (based on motor thermal model) has exceeded fault limit defined by parameter 31.04 Mot temp1 fltLim .	<p>Check motor ratings and load.</p> <p>Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc.</p> <p>Check value of fault limit.</p> <p>Check motor thermal model settings (parameters 31.09...31.14).</p>
		Measured motor temperature has exceeded fault limit defined by parameter 31.04 Mot temp1 fltLim .	<p>Check that actual number of sensors corresponds to value set by parameter 31.02 Mot temp1 src.</p> <p>Check motor ratings and load.</p> <p>Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc.</p> <p>Check value of fault limit.</p>
0049	AI SUPERVISION (0x8110) Programmable fault: 13.32 AI superv func	An analogue input has reached limit defined by parameter 13.33 AI superv cw .	<p>Check analogue input source and connections.</p> <p>Check analogue input minimum and maximum limit settings.</p>

Code	Fault (fieldbus code)	Cause	What to do
0050	ENCODER 1 CABLE (0x7389) Programmable fault: 90.05 Enc cable fault	Encoder 1 cable fault detected.	Check cable between FEN-xx interface and encoder 1. After any modifications in cabling, re-configure interface by switching drive power off and on, or by activating parameter 90.10 Enc par refresh .
0051	ENCODER 2 CABLE (0x738A) Programmable fault: 90.05 Enc cable fault	Encoder 2 cable fault detected.	Check cable between FEN-xx interface and encoder 2. After any modifications in cabling, re-configure interface by switching drive power off and on, or by activating parameter 90.10 Enc par refresh .
0052	D2D CONFIG (0x7583)	Configuration of the drive-to-drive link has failed for a reason other than those indicated by alarm A- 2042 , for example start inhibition is requested but not granted.	Contact your local ABB representative.
0053	D2D COMM (0x7520) Programmable fault: 57.02 Comm loss func	On the master drive: The drive has not been replied to by an activated follower for five consecutive polling cycles.	Check that all drives that are polled (parameters 57.04 Follower mask 1 and 57.05 Follower mask 2) on the drive-to-drive link are powered, properly connected to the link, and have the correct node address. Check the drive-to-drive link wiring.
		On a follower drive: The drive has not received new reference 1 and/or 2 for five consecutive reference handling cycles.	Check the settings of parameters 57.06 Ref 1 src and 57.07 Ref 2 src on the master drive. Check the drive-to-drive link wiring.
0054	D2D BUF OVLOAD (0x7520) Programmable fault: 90.05 Enc cable fault	Transmission of drive-to-drive references failed because of message buffer overflow.	Contact your local ABB representative.
0055	TECH LIB (0x6382)	Resettable fault generated by a technology library.	Refer to the documentation of the technology library.
0056	TECH LIB CRITICAL (0x6382)	Permanent fault generated by a technology library.	Refer to the documentation of the technology library.
0057	FORCED TRIP (0xFF90)	Generic Drive Communication Profile trip command.	Check PLC status.
0058	FB PAR ERROR (0x6320)	The drive does not have a functionality requested by PLC, or requested functionality has not been activated.	Check PLC programming. Check settings of parameter group 50 Fieldbus .
0059	STALL (0x7121) Programmable fault: 30.09 Stall function	Motor is operating in stall region because of e.g. excessive load or insufficient motor power.	Check motor load and drive ratings. Check fault function parameters.
0060	LOAD CURVE (0x2312) Programmable fault: 34.01 Overload func / 34.02 Underload func	Overload or underload limit has been exceeded.	Check the settings of the parameters in group 34 User load curve .

Code	Fault (fieldbus code)	Cause	What to do
0061	SPEED FEEDBACK (0x8480)	No speed feedback is received.	Check the settings of the parameters in group 19 Speed calculation . Check encoder installation. See the description of fault 0039 (ENCODER1) for more information.
0062	D2D SLOT COMM (0x7584)	Drive-to-drive link is set to use an FMBA module for communication, but no module is detected in specified slot.	Check the settings of parameters 57.01 and 57.15 . Ensure that the FMBA module has been detected by checking parameters 09.20...09.22 . Check that the FMBA module is correctly wired. Try installing the FMBA module into another slot. If the problem persists, contact your local ABB representative.
0063	MOTOR TEMP2 (0x4313) Programmable fault: 31.05 Mot temp2 prot	Estimated motor temperature (based on motor thermal model) has exceeded fault limit defined by parameter 31.08 Mot temp2 fltLim .	Check motor ratings and load. Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc. Check value of alarm limit. Check motor thermal model settings (parameters 31.09...31.14).
		Measured motor temperature has exceeded fault limit defined by parameter 31.08 Mot temp2 fltLim .	Check that actual number of sensors corresponds to value set by parameter 31.06 Mot temp2 src . Check motor ratings and load. Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc. Check value of alarm limit.
0064	IGBT OVERLOAD (0x5482)	Excessive IGBT junction to case temperature. This fault protects the IGBT(s) and can be activated by a short circuit in the motor cable.	Check motor cable.
0065	IGBT TEMP (0x4210)	Drive IGBT temperature is excessive.	Check ambient conditions. Check air flow and fan operation. Check heatsink fins for dust pick-up. Check motor power against drive power.
0066	COOLING (0x4290)	Drive module temperature is excessive.	Check setting of parameter 95.03 Temp inu ambient . Check ambient temperature. If it exceeds 40 °C (104 °F), ensure that load current does not exceed derated load capacity of drive. See the appropriate <i>Hardware Manual</i> . Check drive module cooling air flow and fan operation. Check inside of cabinet and heatsink of drive module for dust pick-up. Clean whenever necessary.
0067	FPGA ERROR (0x5401)	Drive internal fault	Contact your local ABB representative.
0068	FPGA ERROR2 (0x5402)	Drive internal fault	Contact your local ABB representative.

300 Fault tracing

Code	Fault (fieldbus code)	Cause	What to do
0069	ADC ERROR (0x5403)	Drive internal fault	Contact your local ABB representative.
0070	TEMP MEAS FAILURE (0x4211)	Problem with internal temperature measurement of the drive.	Contact your local ABB representative.
0071	EFB COMM LOSS FAULT (0x7540)	Embedded fieldbus interface has been taken into use, and there is a communication break between the drive and the master station.	Check: <ul style="list-style-type: none"> • selection of the parameter which enables/disables EFB communication (58.01 Protocol ena sel) • EFB connection at terminal XD2D on the JCON board • status of the fieldbus master (online/offline) • settings of the communication supervision function (parameter 58.09 Comm loss action).
0072	TEMP DIFFERENCE (0x4212)	Too high a temperature difference between the IGBTs of different phases.	Check cooling and the fan. Contact your local ABB representative
0201	T2 OVERLOAD (0x0201)	Firmware time level 2 overload Note: This fault cannot be reset.	Contact your local ABB representative.
0202	T3 OVERLOAD (0x6100)	Firmware time level 3 overload Note: This fault cannot be reset.	Contact your local ABB representative.
0203	T4 OVERLOAD (0x6100)	Firmware time level 4 overload Note: This fault cannot be reset.	Contact your local ABB representative.
0204	T5 OVERLOAD (0x6100)	Firmware time level 5 overload Note: This fault cannot be reset.	Contact your local ABB representative.
0205	A1 OVERLOAD (0x6100)	Application time level 1 fault Note: This fault cannot be reset.	Contact your local ABB representative.
0206	A2 OVERLOAD (0x6100)	Application time level 2 fault Note: This fault cannot be reset.	Contact your local ABB representative.
0207	A1 INIT FAULT (0x6100)	Application task creation fault Note: This fault cannot be reset.	Contact your local ABB representative.
0208	A2 INIT FAULT (0x6100)	Application task creation fault Note: This fault cannot be reset.	Contact your local ABB representative.
0209	STACK ERROR (0x6100)	Drive internal fault Note: This fault cannot be reset.	Contact your local ABB representative.
0210	FPGA ERROR (0xFF61)	Drive internal fault Note: This fault cannot be reset.	Contact your local ABB representative.

Code	Fault (fieldbus code)	Cause	What to do
0301	UFF FILE READ (0x6300)	File read error Note: This fault cannot be reset.	Contact your local ABB representative.
0302	APPL DIR CREATION (0x6100)	Drive internal fault Note: This fault cannot be reset.	Contact your local ABB representative.
0303	FPGA CONFIG DIR (0x6100)	Drive internal fault Note: This fault cannot be reset.	Contact your local ABB representative.
0304	PU RATING ID (0x5483)	Drive internal fault Note: This fault cannot be reset.	Contact your local ABB representative.
0305	RATING DATABASE (0x6100)	Drive internal fault Note: This fault cannot be reset.	Contact your local ABB representative.
0306	LICENSING (0x6100)	Drive internal fault Note: This fault cannot be reset.	Contact your local ABB representative.
0307	DEFAULT FILE (0x6100)	Drive internal fault Note: This fault cannot be reset.	Contact your local ABB representative.
0308	APPLFILE PAR (0x6300)	Corrupted application file Note: This fault cannot be reset.	Reload application. If fault is still active, contact your local ABB representative.
0309	APPL LOADING (0x6300)	Application file incompatible or corrupted Note: This fault cannot be reset.	Check the fault logger for a fault code extension. See appropriate actions for each extension below. <i>*Refer to Application programming for ACS850 drives (3AUA0000078664 [English]).</i>
	Extension: 8	Template used in the application incompatible with drive firmware.	<i>*Change the template of the application in DriveSPC.</i>
	Extension: 10	Parameters defined in the application conflict with existing drive parameters.	<i>*Check the application for conflicting parameters.</i>
	Extension: 35	Application memory full.	Contact your local ABB representative.
	Extension: Other	Corrupted application file	<i>*Reload application.</i> If fault is still active, contact your local ABB representative.
0310	USERSET LOAD (0xFF69)	Loading of user set is not successfully completed because: - requested user set does not exist - user set is not compatible with drive program - drive has been switched off during loading.	Reload.

302 Fault tracing

Code	Fault (fieldbus code)	Cause	What to do
0311	USERSET SAVE (0xFF69)	User set is not saved because of memory corruption.	Check the setting of parameter 95.01 Ctrl boardSupply . If the fault still occurs, contact your local ABB representative.
0312	UFF OVERSIZE (0x6300)	UFF file is too big	Contact your local ABB representative.
0313	UFF EOF (0x6300)	UFF file structure failure	Contact your local ABB representative.
0314	TECH LIB INTERFACE (0x6100)	Incompatible firmware interface Note: This fault cannot be reset.	Contact your local ABB representative.
0315	RESTORE FILE (0x630D)	Restoration of backed-up parameters failed	Contact your local ABB representative.
0316	DAPS MISMATCH (0x5484)	Mismatch between JCU Control Unit firmware and power unit logic versions.	Contact your local ABB representative.
0317	SOLUTION FAULT (0x6200)	Fault generated by function block SOLUTION_FAULT in the application program.	Check the usage of the SOLUTION_FAULT block in the application program.
0318	MENU HIDING (0x6200)	Menu hiding file missing or corrupted.	Reload application. Contact your local ABB representative.



Control through the embedded fieldbus interface

efesotomasyon.com

What this chapter contains

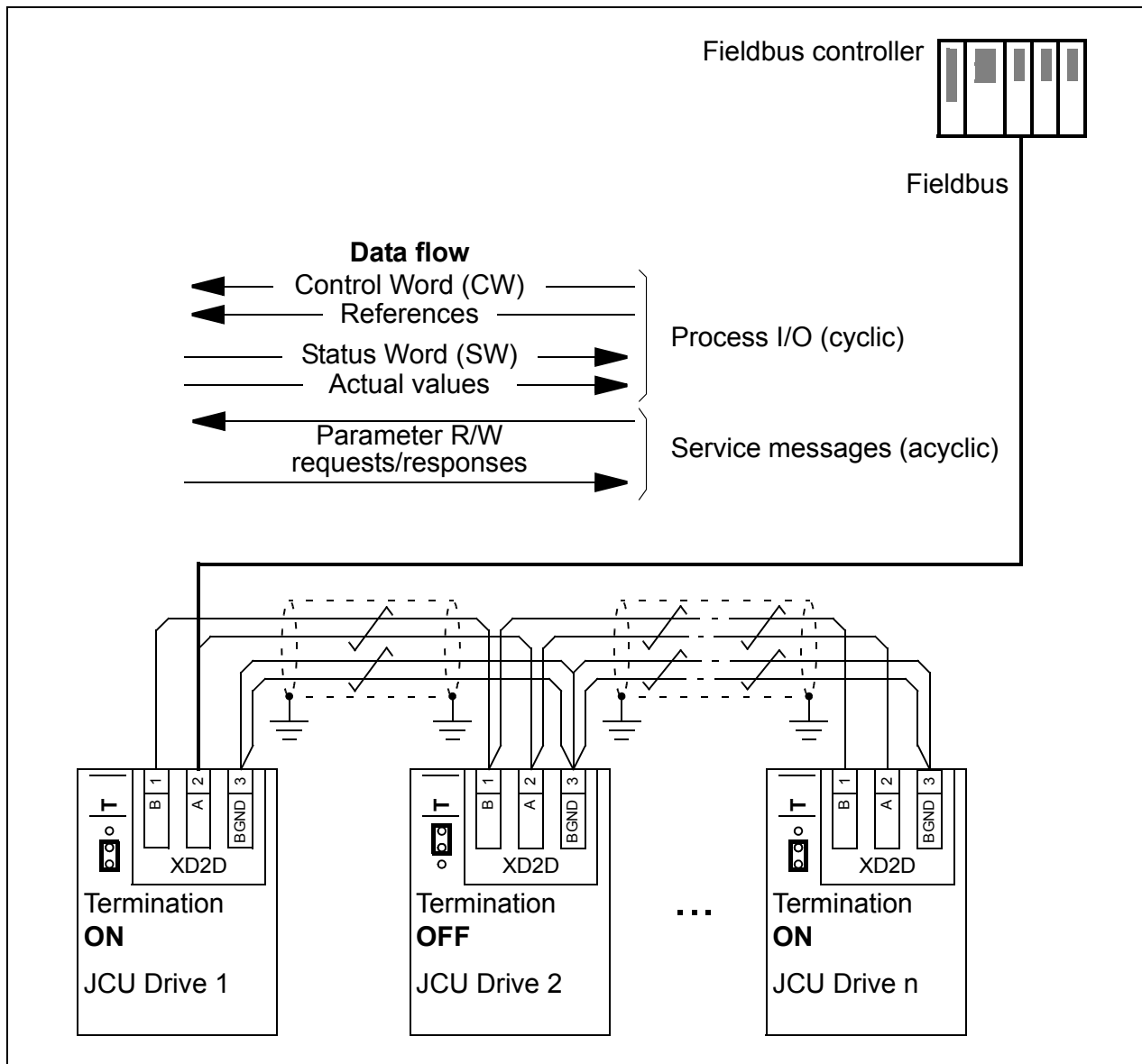
The chapter describes how the drive can be controlled by external devices over a communication network (fieldbus) using an embedded fieldbus interface.

System overview

The drive can be connected to an external control system through a serial communication link using either a fieldbus adapter or an embedded fieldbus interface.

The embedded fieldbus interface supports the Modbus RTU protocol. The drive control program can receive and send cyclic data from and to the Modbus master on 10 ms time level. The actual communication speed depends on other factors as well, such as the baud rate (a parameter setting in the drive).

The drive can be set to receive all of its control information through the fieldbus interface, or the control can be distributed between the fieldbus interface and other available sources, for example, digital and analogue inputs.



Connecting the embedded fieldbus to the drive

Connect the embedded fieldbus interface to terminal XD2D on the JCON board of the drive. See the appropriate *Hardware Manual* for more information on the connection, chaining and termination of the link.

XD2D is the connection point for a drive-to-drive link, a daisy-chained RS-485 transmission line with one master and multiple slaves.

Note: If the XD2D connector is used for the embedded fieldbus interface (parameter [58.01 Protocol ena sel](#) is set to *Modbus RTU*), the drive-to-drive link operation (parameter group 57) is automatically disabled.

Setting up the embedded fieldbus interface

Set the drive up for the embedded fieldbus communication with the parameters shown in the table below. The **Setting for fieldbus control** column gives either the value to use or the default value. The **Function/Information column** gives a description of the parameter or instructs in its use.

Parameter	Setting for fieldbus control	Function/Information
COMMUNICATION INITIALIZATION		
50.15 <i>Fb cw used</i>	P.02.36	Selects the address of the fieldbus Control Word in use (02.36 EFB main cw).
58.01 <i>Protocol enable</i>	<i>Modbus RTU</i> (default)	Initializes embedded fieldbus communication. Drive-to-drive link operation (parameter group 57) is automatically disabled.
EMBEDDED MODBUS CONFIGURATION		
58.03 <i>Node address</i>	1 (default)	Node address. There may not be two nodes with the same node address online.
58.04 <i>Baud rate</i>	9600 (default)	Defines the communication speed of the link. Use the same setting as in the master station.
58.05 <i>Parity</i>	<i>8 none 1</i> (default)	Selects the parity and stop bit setting. Use the same setting as in the master station.
58.06 <i>Control profile</i>	<i>ABB Enhanced</i> (default)	Selects the communication profile used by the drive. See section Basics of the embedded fieldbus interface on page 310.
58.07 <i>Comm loss timeout</i>	600 (default)	Defines the timeout limit for the EFB communication monitoring.
58.08 <i>Comm loss mode</i>	<i>None</i> (default)	Enables/disables EFB communication loss monitoring and defines the means for resetting the counter of the communication loss delay.
58.09 <i>Comm loss action</i>	<i>None</i> (default)	Defines the drive operation after the EFB communication loss monitoring awakes.
58.10 <i>Refresh settings</i>	<i>Done</i> (default)	Refreshes the settings of parameters 58.01...58.09 .
58.30 <i>Transmit delay</i>	0 (default)	Defines the delay time which the slave waits until it sends a response.
58.31 <i>Ret app errors</i>	<i>Yes</i> (default)	Selects whether the drive returns Modbus exception codes or not.
58.32 <i>Word order</i>	<i>LSW MSW</i> (default)	Defines the order of the data words in the Modbus frame.

Parameter	Setting for fieldbus control	Function/Information
58.35 <i>Data I/O 1</i> ... 58.58 <i>Data I/O 24</i>	0 (default)	Defines the address of the drive parameter which the Modbus master accesses when it reads from or writes to the register address corresponding to Modbus In/Out parameters. Select the parameters that you want to read or write through the Modbus I/O words.

The new settings will take effect when the drive is powered up the next time, or when parameter [58.10 Refresh settings](#) is activated.

Setting the drive control parameters

After the embedded fieldbus interface has been set up, check and adjust the drive control parameters listed in the table below. The **Setting for fieldbus control** column gives the value or values to use when the embedded fieldbus signal is the desired source or destination for that particular drive control signal. The **Function/Information** column gives a description of the parameter.

Parameter	Setting for fieldbus control	Function/Information
-----------	------------------------------	----------------------

CONTROL COMMAND SOURCE SELECTION		
10.01 Ext1 start func	FB	Selects fieldbus as the source for the start and stop commands when EXT1 is selected as the active control location.
10.04 Ext2 start func	FB	Selects fieldbus as the source for the start and stop commands when EXT2 is selected as the active control location.
10.10 Fault reset sel	P.02.36.08	Selects the fault reset bit of signal 02.36 EFB main cw as the source for the fault reset command of the drive.
Note: To start and stop the drive through control location EXT1, set parameter 10.01 to FB and keep parameter 12.01 to its default value (C.FALSE).		

SPEED REFERENCE SELECTION		
21.01 Speed ref1 sel	EFB ref1 or EFB ref2	Selects a reference received through the embedded fieldbus interface as the speed reference ref1 of the drive.
21.02 Speed ref2 sel	EFB ref1 or EFB ref2	Selects a reference received through the embedded fieldbus interface as the speed reference ref2 of the drive.
Note: To control the drive speed with fieldbus reference REF1, set parameter 21.01 to EFB ref1 , and keep parameters 12.03 and 21.04 to their default values (Speed and C.FALSE).		

TORQUE REFERENCE SELECTION		
24.01 Torq ref1 sel	EFB ref1 or EFB ref2	Selects one of the references received through the embedded fieldbus interface as the torque reference ref1 of the drive.
24.02 Torq ref add sel	EFB ref1 or EFB ref2	Selects one of the references received through the embedded fieldbus interface as the torque reference ref2 of the drive.
Note: To control the torque of the drive with fieldbus reference REF2, set parameter 24.01 to EFB ref2 , keep 12.01 to its default value (C.FALSE) and set 12.03 to Torque.		

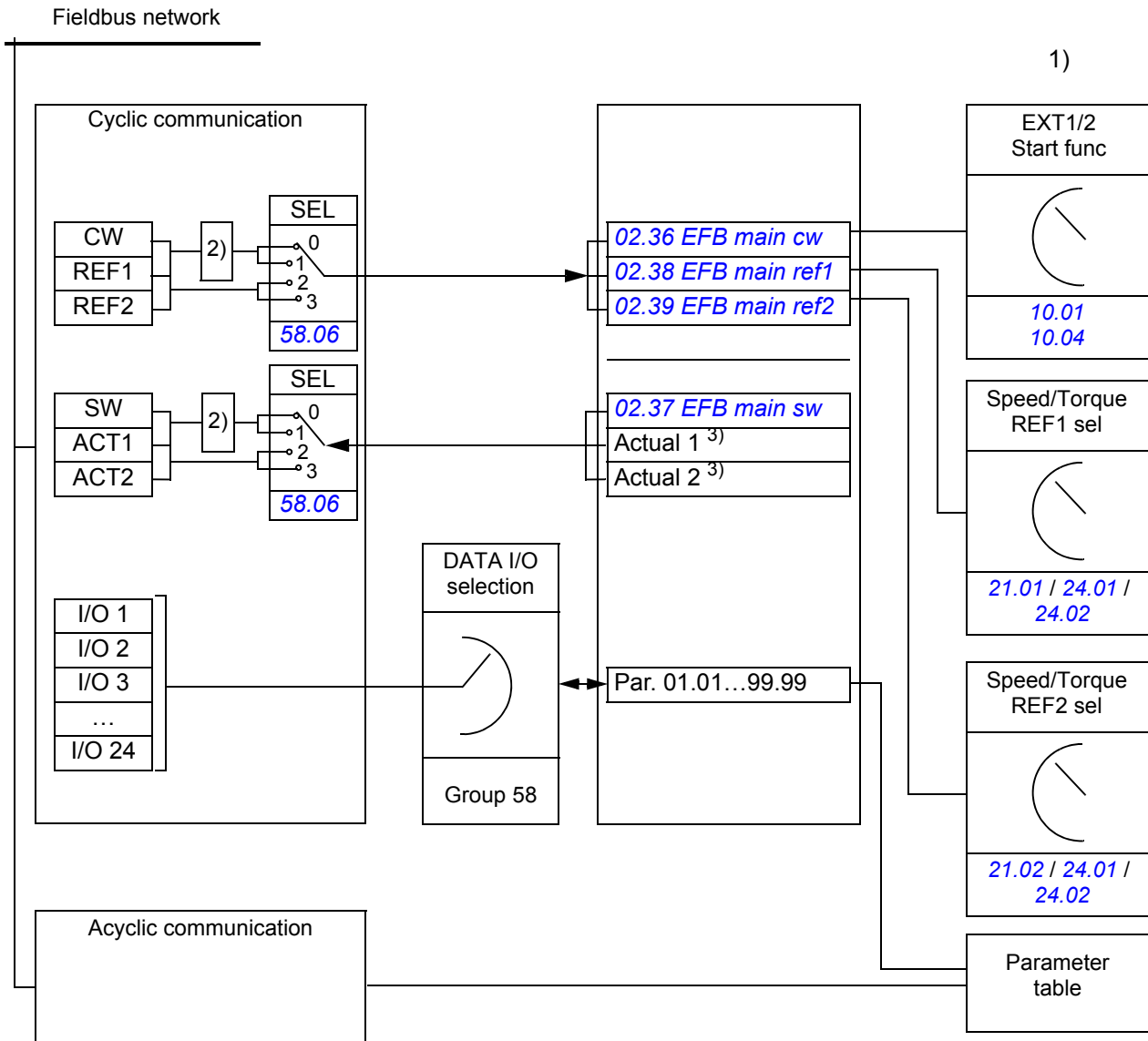
REFERENCE SCALING		
50.04 Fb ref1 modesel	Raw data Torque Speed	Defines the fieldbus reference REF1 scaling. Selects also the fieldbus actual signal act1 when set to Torque or Speed .

Parameter	Setting for fieldbus control	Function/Information
<i>50.05 Fb ref2 modesel</i>	<i>Raw data</i> <i>Torque</i> <i>Speed</i>	Defines the fieldbus reference REF2 scaling. Selects also the fieldbus actual signal act2 when set to <i>Torque</i> or <i>Speed</i> .
ACTUAL VALUE ACT1 AND ACT 2 SELECTION (if <i>50.04</i> or <i>50.05</i> has value <i>Raw data</i>).		
<i>50.06 Fb act1 tr src</i>	Any	Selects the source for fieldbus actual value act1 when parameter <i>50.04 Fb ref1 modesel</i> is set to <i>Raw data</i> .
<i>50.07 Fb act2 tr src</i>	Any	Selects the source for fieldbus actual value act2 when parameter <i>50.05 Fb ref2 modesel</i> is set to <i>Raw data</i> .
SYSTEM CONTROL INPUTS		
<i>16.07 Param save</i>	<i>Save</i> (restores to <i>Done</i>)	Saves parameter value changes (including those made through fieldbus control) to permanent memory.

Basics of the embedded fieldbus interface

The cyclic communication between a fieldbus system and the drive consists of 16-bit data words (with the ABB Drives profile or DCU 16-bit profile) or 32-bit data words (with the DCU 32-bit profile).

The diagram below illustrates the operation of the embedded fieldbus interface. The signals transferred in the cyclic communication are explained further below the diagram.



- 1) See also other parameters which can be controlled by the fieldbus.
- 2) Data conversion if parameter 58.06 Control profile is ABB Classic or ABB Enhanced. See section *About the EFB communication profiles* on page 312.
- 3) See parameter 50.01 Fb ref1 modesel and 50.02 Fb ref2 modesel for the actual value selections.

■ Control Word and Status Word

The fieldbus Control Word (CW) is a 16-bit or 32-bit packed boolean word. It is the principal means of controlling the drive from a fieldbus system. The CW is sent by the fieldbus controller to the drive. The drive switches between its states according to the bit-coded instructions of the CW. In the embedded fieldbus communication, the CW is written to drive parameter [02.36 EFB main cw](#) from where you can use it in the control of the drive. The fieldbus CW is either written to the drive CW as it is, or the data is converted. See section [About the EFB communication profiles](#) on page [312](#).

The fieldbus Status Word (SW) is a 16-bit or 32-bit packed boolean word. It contains status information from the drive to the fieldbus controller. In the embedded fieldbus communication, the SW is read from drive parameter [02.37 EFB main sw](#). The drive SW is either written to the fieldbus SW as it is or the data is converted. See section [About the EFB communication profiles](#) on page [312](#).

■ References

Fieldbus references (REF1 and REF2) are 16-bit or 32-bit signed integers. The contents of each reference word can be used as the speed, frequency, torque or process reference. In the embedded fieldbus communication, the REF1 and REF2 are written to [02.38 EFB main ref1](#) and [02.39 EFB main ref2](#) from where you can use them in the control of the drive. The references are either written to the drive references as they are, or the values are scaled. See section [About the EFB communication profiles](#) on page [312](#).

■ Actual values

Fieldbus actual signals (ACT1 and ACT2) are 16-bit or 32-bit signed integers. They convey selected drive parameter values from the drive to the master. The drive values are either written to the fieldbus actual values as they are, or the values are scaled. See section [About the EFB communication profiles](#) on page [312](#).

■ Data inputs/outputs

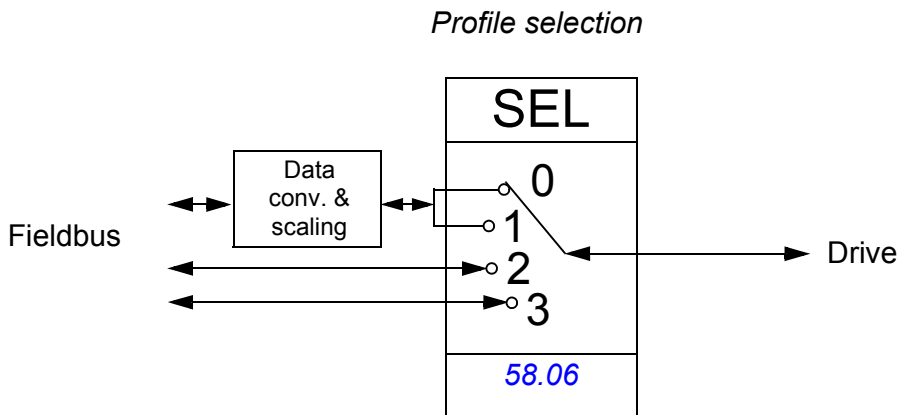
Data input/output (I/O) are 16-bit or 32-bit words containing selected drive parameter values. Parameters [58.35 Data I/O 1](#) ... [58.58 Data I/O 24](#) define the addresses from which the master either reads data (input) or to which it writes data (output).

About the EFB communication profiles

A communication profile defines the rules for data transfer between the drive and the fieldbus master, for example:

- if packed boolean words are converted and how
- if signal values are scaled and how
- how drive register addresses are mapped for the fieldbus master.

You can configure the drive to receive and send messages according to one of the four profiles: the ABB Drives classic profile, ABB Drives enhanced profile, DCU 16-bit profile or DCU 32-bit profile. For either one of the ABB Drives profile, the embedded fieldbus interface of the drive converts the fieldbus data to and from the native data used in the drive. Both DCU profiles are transparent, that is, no data conversion is done. The figure below illustrates the effect of the profile selection.



Communication profile selection with parameter *58.06 Control profile* is:

- *ABB Classic*
- *ABB Enhanced*
- *DCU 16-bit*
- *DCU 32-bit*

ABB Drives classic profile and ABB Drives enhanced profile

■ Control Word for the ABB Drives profiles

The table below shows the contents of the fieldbus Control Word for both ABB Drives profiles. The embedded fieldbus interface converts this word to the form in which it is used in the drive (*02.36 EFB main cw*). The upper case boldface text refers to the states shown in *State transition diagram for the ABB Drives profiles* on page 317.

Bit	Name	Value	STATE/Description
0	OFF1_ CONTROL	1	Proceed to READY TO OPERATE .
		0	Stop along currently active deceleration ramp. Proceed to OFF1 ACTIVE ; proceed to READY TO SWITCH ON unless other interlocks (OFF2, OFF3) are active.
1	OFF2_ CONTROL	1	Continue operation (OFF2 inactive).
		0	Emergency OFF, coast to stop. Proceed to OFF2 ACTIVE , proceed to SWITCH-ON INHIBITED .
2	OFF3_ CONTROL	1	Continue operation (OFF3 inactive).
		0	Emergency stop, stop within time defined by drive parameter. Proceed to OFF3 ACTIVE ; proceed to SWITCH-ON INHIBITED . Warning: Ensure that the motor and driven machine can be stopped using this stop mode.
3	INHIBIT_ OPERATION	1	Proceed to OPERATION ENABLED . Note: Run enable signal must be active; see the drive documentation. If the drive is set to receive the Run enable signal from the fieldbus, this bit activates the signal.
		0	Inhibit operation. Proceed to OPERATION INHIBITED .
4	RAMP_OUT_ ZERO	1	Normal operation. Proceed to RAMP FUNCTION GENERATOR: OUTPUT ENABLED .
		0	Force Ramp Function Generator output to zero. Drive ramps to stop (current and DC voltage limits in force).
5	RAMP_HOLD	1	Enable ramp function. Proceed to RAMP FUNCTION GENERATOR: ACCELERATOR ENABLED .
		0	Halt ramping (Ramp Function Generator output held).
6	RAMP_IN_ ZERO	1	Normal operation. Proceed to OPERATING . Note: This bit is effective only if the fieldbus interface is set as the source for this signal by drive parameters.
		0	Force Ramp Function Generator input to zero.

314 Control through the embedded fieldbus interface

Bit	Name	Value	STATE/Description
7	RESET	0=>1	Fault reset if an active fault exists. Proceed to SWITCH-ON INHIBITED . Note: This bit is effective only if the fieldbus interface is set as the source for this signal by drive parameters.
		0	Continue normal operation.
8, 9	Reserved.		
10	REMOTE_CMD	1	Fieldbus control enabled.
		0	Control Word <> 0 or Reference <> 0: Retain last Control Word and Reference. Control Word = 0 and Reference = 0: Fieldbus control enabled. Reference and deceleration/acceleration ramp are locked.
11	EXT_CTRL_LOC	1	Select External Control Location EXT2. Effective if the control location is parameterized to be selected from the fieldbus.
		0	Select External Control Location EXT1. Effective if the control location is parameterized to be selected from the fieldbus.
12 ...15	Reserved		

■ Status Word for the ABB Drives profiles

The table below shows the fieldbus Status Word for both ABB Drives profiles. The embedded fieldbus interface converts the drive Status Word (*02.37 EFB main sw*) to this form for the transfer in the fieldbus. The upper case boldface text refers to the states shown in *State transition diagram for the ABB Drives profiles* on page 317.

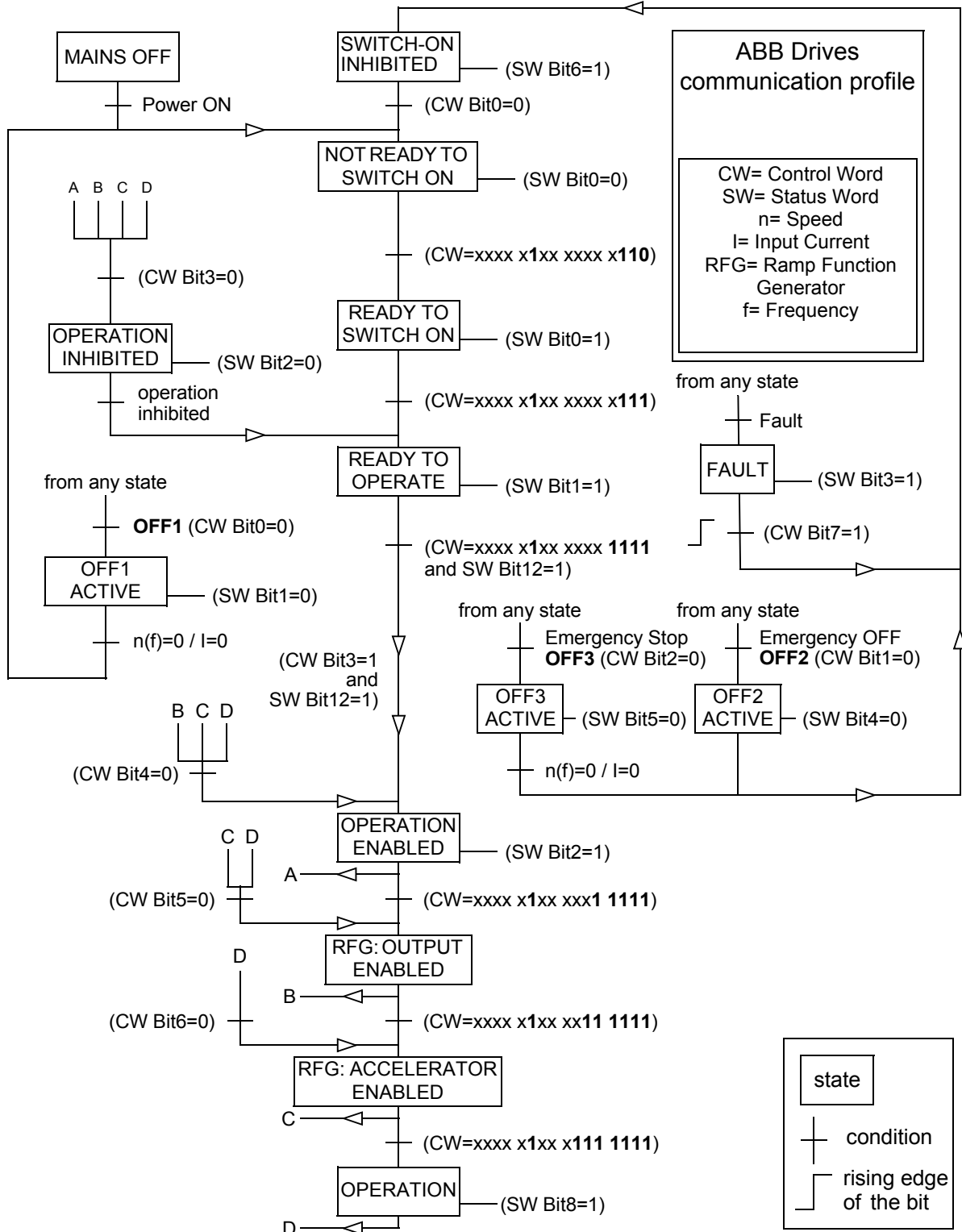
Bit	Name	Value	STATE/Description
0	RDY_ON	1	READY TO SWITCH ON.
		0	NOT READY TO SWITCH ON.
1	RDY_RUN	1	READY TO OPERATE.
		0	OFF1 ACTIVE.
2	RDY_REF	1	OPERATION ENABLED.
		0	OPERATION INHIBITED.
3	TRIPPED	1	FAULT.
		0	No fault.
4	OFF_2_STA	1	OFF2 inactive.
		0	OFF2 ACTIVE.
5	OFF_3_STA	1	OFF3 inactive.
		0	OFF3 ACTIVE.
6	SWC_ON_ INHIB	1	SWITCH-ON INHIBITED.
		0	–
7	ALARM	1	Warning/Alarm.
		0	No warning/alarm.
8	AT_ SETPOINT	1	OPERATING. Actual value equals Reference = is within tolerance limits, i.e. in speed control, speed error is 10% max. of nominal motor speed.
		0	Actual value differs from Reference = is outside tolerance limits.
9	REMOTE	1	Drive control location: REMOTE (EXT1 or EXT2).
		0	Drive control location: LOCAL.
10	ABOVE_ LIMIT	1	Actual frequency or speed equals or exceeds supervision limit (set by drive parameter). Valid in both directions of rotation.
		0	Actual frequency or speed within supervision limit.
11	EXT_CTRL_ LOC	1	External Control Location EXT2 selected.
		0	External Control Location EXT1 selected.
12	EXT_RUN_ ENABLE	1	External Run Enable signal received.
		0	No External Run Enable signal received.
13 ... 14	Reserved		

316 *Control through the embedded fieldbus interface*

Bit	Name	Value	STATE/Description
15		1	Communication error detected by fieldbus adapter module.
		0	Fieldbus adapter communication OK.

■ State transition diagram for the ABB Drives profiles

The diagram below shows the state transitions in the drive when the drive has either one of the ABB Drives profiles in use and the drive is configured to follow the commands of the fieldbus Control Word. The upper case texts refer to the states which are used in the tables representing the fieldbus Control and Status words. See sections [Control Word for the ABB Drives profiles](#) on page 313 and [Status Word for the ABB Drives profiles](#) on page 315.



■ References for the ABB Drives profiles

The ABB Drives profiles support the use of two fieldbus references, REF1 and REF2. The references are 16-bit words each containing a sign bit and a 15-bit integer. A negative reference is formed by calculating the two's complement from the corresponding positive reference.

The fieldbus references are scaled before they are written into signals [02.38 EFB main ref1](#) or [02.39 EFB main ref2](#) for the use in the drive. Parameters [50.04 Fb ref1 modesel](#) and [50.05 Fb ref2 modesel](#) define the scaling and possible use of the fieldbus reference REF1 and REF2 as follows:

- If you select value [Speed](#), the fieldbus reference can be used as a speed reference and it is scaled as follows:

Fieldbus reference REF1 or REF2 [integer]	Corresponding speed reference in the drive [rpm]
20 000	value of parameter 19.01 Speed scaling
0	0
-20 000	-(value of parameter 19.01 Speed scaling)

- If you select value [Torque](#), the fieldbus reference can be used as a torque reference and it is scaled as follows:

Fieldbus reference REF1 or REF2 [integer]	Corresponding torque reference in the drive [%]
10 000	100% of motor nominal torque
0	0
-10 000	-(100% of motor nominal torque)

- If you select value [Raw data](#), the fieldbus reference REF1 or REF2 is the drive reference without scaling.

Fieldbus reference REF1 or REF2 [integer]	Corresponding reference in the drive [rpm or %] ¹⁾
32 767	32 767
0	0
-32 768	-32 768

¹⁾ Unit depends on the use of the reference in the drive. Rpm for speed reference and % for torque.

■ Actual values for the ABB Drives profiles

Both the ABB Drives classic profile and ABB Drives enhanced profile support the use of two fieldbus actual values, ACT1 and ACT2. The actual values are 16-bit words each containing a sign bit and a 15-bit integer. A negative value is formed by calculating the two's complement from the corresponding positive value.

The drive signals are scaled before they are written into fieldbus actual values, ACT1 and ACT2. Parameters [50.04 Fb ref1 modesel](#) and [50.05 Fb ref2 modesel](#) both select the drive actual signals and define the scaling as follows:

- If you select value [Speed](#), drive actual signal [01.01 Motor speed rpm](#) is scaled and written to the fieldbus actual value. The table below shows the scaling:

Value of 01.01 Motor speed rpm [rpm]	Corresponding fieldbus actual value ACT1 or ACT2 [integer]
value of parameter 19.01 Speed scaling	20 000
0	0
-(value of parameter 19.01 Speed scaling)	-20 000

- If you select value [Torque](#), drive actual signal [01.06 Motor torque](#) is scaled and written to the fieldbus actual value. The table below shows the scaling:

Value of 01.06 Motor torque [%]	Corresponding fieldbus actual value ACT1 or ACT2 [integer]
100% of motor nominal torque	10 000
0	0
-(100% of motor nominal torque)	-10 000

- If you select value [Raw data](#), the fieldbus actual value ACT1 or ACT2 is the drive actual value without scaling.

Drive value	Corresponding fieldbus actual value ACT1 or ACT2 [integer]
32 767	32 767
0	0
-32 768	-32 768

■ Modbus register addresses for the ABB Drives classic profile

The table below shows the Modbus register addresses for the drive data with the ABB Drives classic profile. This profile provides a converted 16-bit access to the drive data.

Note: Only the least significant 16-bits of the drive 32-bit Control and Status words can be accessed.

Register Address	Register Data (16-bit)
400001	Fieldbus Control Word (CW). See section Control Word for the ABB Drives profiles on page 313.
400002	Fieldbus reference 1 (REF1)
400003	Fieldbus reference 2 (REF2)
400004	Fieldbus Status Word (SW). See section Status Word for the ABB Drives profiles on page 315.
400005	Fieldbus actual value 1 (ACT1)
400006	Fieldbus actual value 2 (ACT2)
400007	Fieldbus data in/out 1 (Drive parameter 58.35 Data I/O 1)
...	...
400030	Fieldbus data in/out 24 (Drive Parameter 58.58 Data I/O 24)
400101...409999	<p>Register address (16-bit drive parameter) = 400000 + 100 × group + index</p> <p>Example: Modbus register address to drive parameter 03.18 is $400000 + 100 \times 3 + 18 = 400318$</p> <p>Drive parameter access (32-bit drive parameter) = 420000 + 200 × group + 2 × index</p> <p>Example: Modbus register address to drive parameter 01.27 $420000 + 200 \times 1 + 2 \times 27 = 420254$</p>

■ Modbus register addresses for the ABB Drives enhanced profile

Register address	Register data (16-bit words)
400001	Fieldbus Control Word (CW). See section Control Word for the ABB Drives profiles on page 313.
400002	Fieldbus reference 1 (REF1).
400003	Fieldbus reference 2 (REF2)
400004	Fieldbus data in/out 1 (Drive parameter 58.35 Data I/O 1)
...	...
400015	Fieldbus data in/out 12 (Drive parameter 58.46 Data I/O 12)
400051	Fieldbus Status Word (SW). See section Status Word for the ABB Drives profiles on page 315.
400052	Fieldbus actual value 1 (ACT1)
400053	Fieldbus actual value 2 (ACT2)
400054	Fieldbus data in/out 13 (Drive parameter 58.47 Data I/O 12)
...	...
400065	Fieldbus data in/out 24 (Drive parameter 58.58 Data I/O 24)
400101...409999	<p>Register address (16-bit drive parameter) = 400000 + 100 × group + index</p> <p>Example: Modbus register address to drive parameter 03.18 is $400000 + 100 \times 3 + 18 = 400318$</p> <p>Drive parameter access (32-bit drive parameter) = 420000 + 200 × group + 2 × index</p> <p>Example: Modbus register address to drive parameter 01.27 $420000 + 200 \times 1 + 2 \times 27 = 420254$</p>

DCU 16-bit profile

■ Control and Status words for the DCU 16-bit profile

When the DCU 16-bit profile is in use, the embedded fieldbus interface writes the fieldbus Control Word as is to the drive Control Word bits 0 to 15 (parameter [02.36 EFB main cw](#)). Bits 16 to 32 of the drive Control Word are not in use.

■ Status Word for the DCU 16-bit profile

When the DCU 16-bit profile is in use, the embedded fieldbus interface writes the drive Status Word bits 0 to 15 (parameter [02.37 EFB main sw](#)) to the fieldbus Status Word as is. Bits 16 to 32 of the drive Status Word are not in use.

■ State transition diagram for the DCU 16-bit profile

See section [State diagram](#) on page [339](#) in chapter [Control through a fieldbus adapter](#).

■ References for the DCU 16-bit profile

See section [References for the ABB Drives profiles](#) on page [318](#).

■ Actual signals for the DCU 16-bit profile

See section [Actual values for the ABB Drives profiles](#) on page [319](#).

■ Modbus register addresses for the DCU 16-bit profile

The table below shows the Modbus register addresses and data with the DCU16-bit communication profile.

Note: Only the least significant 16-bits of the drive 32-bit control and status words can be accessed.

Register address	Register data (16-bit)
400001	Control Word (LSW of 02.36 EFB main cw)
400002	Reference 1 (02.38 EFB main ref1)
400003	Reference 2 (02.39 EFB main ref2)
400004	Data in/out 1 (Drive parameter 58.35 Data I/O 1)
...	...
400015	Data in/out 12 (Drive parameter 58.46 Data I/O 12)
400051	Status Word (LSW of 02.37 EFB main sw)
400052	Actual value 1 (selected by parameter 50.01 Fb ref1 modesel)
400053	Actual value 2 (selected by parameter 50.02 Fb ref2 modesel)
400054	Data in/out 13 (drive parameter 58.47 Data I/O 13)
...	...
400065	Data in/out 24 (drive parameter 58.58 Data I/O 24)
400101...409999	<p>Register address (16-bit drive parameter) = 400000 + 100 × group + index</p> <p>Example: Modbus register address to drive parameter 03.18 is $400000 + 100 \times 3 + 18 = 400318$</p> <p>Drive parameter access (32-bit drive parameter) = 420000 + 200 × group + 2 × index</p> <p>Example: Modbus register address to drive parameter 01.27 $420000 + 200 \times 1 + 2 \times 27 = 420254$</p>

DCU 32-bit profile

■ Control and Status words for the DCU 32-bit profile

When the DCU 32-bit profile is in use, the embedded fieldbus interface writes the fieldbus Control Word as is to the drive Control Word (parameter [02.36 EFB main cw](#)).

■ Status word for the DCU 32-bit profile

When the DCU 32-bit profile is in use, the embedded fieldbus interface writes the drive Status Word (parameter [02.37 EFB main sw](#)) as is to the fieldbus Status Word.

■ State transition diagram for the DCU 32-bit profile

See section [State diagram](#) on page [339](#) in chapter [Control through a fieldbus adapter](#).

■ References for the DCU 32-bit profile

The DCU 32-bit profile supports the use of two fieldbus references, REF1 and REF2. The references are 32-bit values consisting of two 16-bit words. The MSW (Most significant word) is the integer part and the LSW (Least significant word) the fractional part of the value. A negative reference is formed by calculating the two's complement from the corresponding positive value of the integer part (MSW).

The fieldbus references are written as is into the drive reference values ([02.38 EFB main ref1](#) or [02.39 EFB main ref2](#)). Parameters [50.04 Fb ref1 modesel](#) and [50.05 Fb ref2 modesel](#) define the reference types (speed or torque) as follows:

- If you select value [Raw data](#), the fieldbus reference type or possible use is not selected. The value is freely usable as a speed or torque reference in the drive.
- If you select value [Speed](#), the fieldbus reference can be used as a speed reference in the drive.
- If you select value [Torque](#), the fieldbus reference can be used as a torque reference in the drive.

The table below clarifies the relation between the fieldbus reference and drive reference (no scaling).

Fieldbus reference REF1 or REF2 [integer and fractional part]	Corresponding reference in the drive [rpm or %] ¹⁾
32767.65535	32767.65535
0	0
-32768.65535	-32768.65535

¹⁾ If the reference value is used as the speed reference, it will be the motor speed in rpm. If the reference value is used as the torque reference, it will be the motor torque in percent of the motor nominal torque.

■ Actual signals for the DCU 32-bit profile

The DCU 32-bit profile supports the use of two fieldbus actual values, ACT1 and ACT2. The actual values are 32-bit values consisting of two 16-bit words. The MSW (Most significant word) is the integer part and the LSW (Least significant word) the fractional part of the 32-bit value. A negative reference is formed by calculating the two's complement from the corresponding positive value of the integer part (MSW).

Parameters [50.04 Fb ref1 modesel](#) and [50.05 Fb ref2 modesel](#) select the drive actual signals for the fieldbus actual values ACT1 and ACT2 respectively as follows:

- If you select value [Raw data](#), drive parameters [50.06 Fb act1 tr src](#) and [50.07 Fb act2 tr src](#) select the drive parameters for the fieldbus actual value ACT1 and ACT2 respectively.
- If you select value [Speed](#), drive parameter [01.01 Motor speed rpm](#) will be written to fieldbus actual value.
- If you select value [Torque](#), drive parameter [01.06 Motor torque](#) will be written to the fieldbus actual value.

The table below clarifies the relation between the value of drive parameter and fieldbus actual value (no scaling).

Value of the selected drive signal	Corresponding fieldbus actual value ACT1 or ACT2 [integer and fractional part]
32767.65535	32767.65535
0	0
-32768.65535	-32768.65535

■ Modbus register addresses for the DCU 32-bit profile

The table below shows the Modbus register addresses and data with the DCU 32-bit profile. This profile provides native 32-bit access to the drive data.

Register address	Register data (16-bit)
400001	Control Word (02.36 EFB main cw) – Least significant 16-bits
400002	Control Word (02.36 EFB main cw) – Most significant 16-bits
400003	Reference 1 (02.38 EFB main ref1) – Least significant 16-bits
400004	Reference 1 (02.38 EFB main ref1) – Most significant 16-bits
400005	Reference 2 (02.39 EFB main ref2) – Least significant 16-bits
400006	Reference 2 (02.39 EFB main ref2) – Most significant 16-bits
400007	Data in/out 1 (Drive parameter 58.35 Data I/O 1)
...	...
400018	Data in/out 12 (Drive parameter 58.46 Data I/O 12)
400051	Status Word (LSW of 02.37 EFB main sw) – Least significant 16-bits
400052	Status Word (MSW of 02.37 EFB main sw) – Most significant 16-bits
400053	Actual value 1 (selected by parameter 50.01 Fb ref1 modesel) – Least significant 16-bits
400054	Actual value 1 (selected by parameter 50.01 Fb ref1 modesel) – Most significant 16-bits
400055	Actual value 2 (selected by parameter 50.02 Fb ref2 modesel) – Least significant 16-bits
400056	Actual value 2 (selected by parameter 50.02 Fb ref2 modesel) – Most significant 16-bits
400057	Data in/out 13 (Drive parameter 58.47 Data I/O 13)
...	...
400068	Data in/out 24 (Drive parameter 58.58 Data I/O 24)
400101...409999	<p>Register address (16-bit drive parameter) = 400000 + 100 × group + index</p> <p>Example: Modbus register address to drive parameter 03.18 is $400000 + 100 \times 3 + 18 = 400318$</p> <p>Drive parameter access (32-bit drive parameter) = 420000 + 200 × group + 2 × index</p> <p>Example: Modbus register address to drive parameter 01.27 $420000 + 200 \times 1 + 2 \times 27 = 420254$</p>

Modbus function codes

Table below shows the Modbus function codes supported by the embedded fieldbus interface.

Code	Function name	Description
0x03	Read Holding Registers	Reads the contents of a contiguous block of holding registers in a server device.
0x06	Write Single Register	Writes a single holding register in a server device.
0x08	Diagnostics	<p>Provides a series of tests for checking the communication between the master and the slave devices, or for checking various internal error conditions within the slave. The following subcodes are supported:</p> <ul style="list-style-type: none"> • 00 Return Query Data: The data passed in the request data field is to be returned in the response. The entire response message should be identical to the request. • 01 Restart Communications Option: The serial line port of the slave device must be initialized and restarted, and all of its communication event counters cleared. If the port is in the Listen Only mode, no response is returned. If the port is not in the Listen Only mode, a normal response is returned before the restart. • 04 Force Listen Only Mode: Forces the addressed slave device to the Listen Only mode. This isolates it from the other devices on the network, allowing them to continue communicating without interruption from the addressed remote device. No response is returned. The only function that will be processed after this mode is entered is the Restart Communications Option function (subcode 01).
0x10	Write Multiple Registers	Writes the contents of a contiguous block of holding registers in a server device.
0x17	Read/Write Multiple Registers	Writes the contents of a contiguous block of holding registers in a server device, then reads the contents of a contiguous block of holding registers (same or different than those written) in a server device.
0x2B/0x0E	Encapsulated Interface Transport / Read Device Identification	<p>Allows reading of identification and other information of the server.</p> <p>Parameter "Read Device ID code" supports one access type:</p> <p>01: Request to get the basic device identification. Returns ABB,ACS850.</p>

Modbus exception codes

Table below shows the Modbus exception codes supported by the embedded fieldbus interface.

Code	Name	Description
0x01	ILLEGAL FUNCTION	The function code received in the query is not an allowable action for the server.
0x02	ILLEGAL DATA ADDRESS	The data address received in the query is not an allowable address for the server.
0x03	ILLEGAL DATA VALUE	A value contained in the query is not an allowable value for the server.
0x04	SLAVE DEVICE FAILURE	An unrecoverable error occurred while the server was attempting to perform the requested action.
0x06	SLAVE DEVICE BUSY	The server is engaged in processing a long-duration program command.



Control through a fieldbus adapter

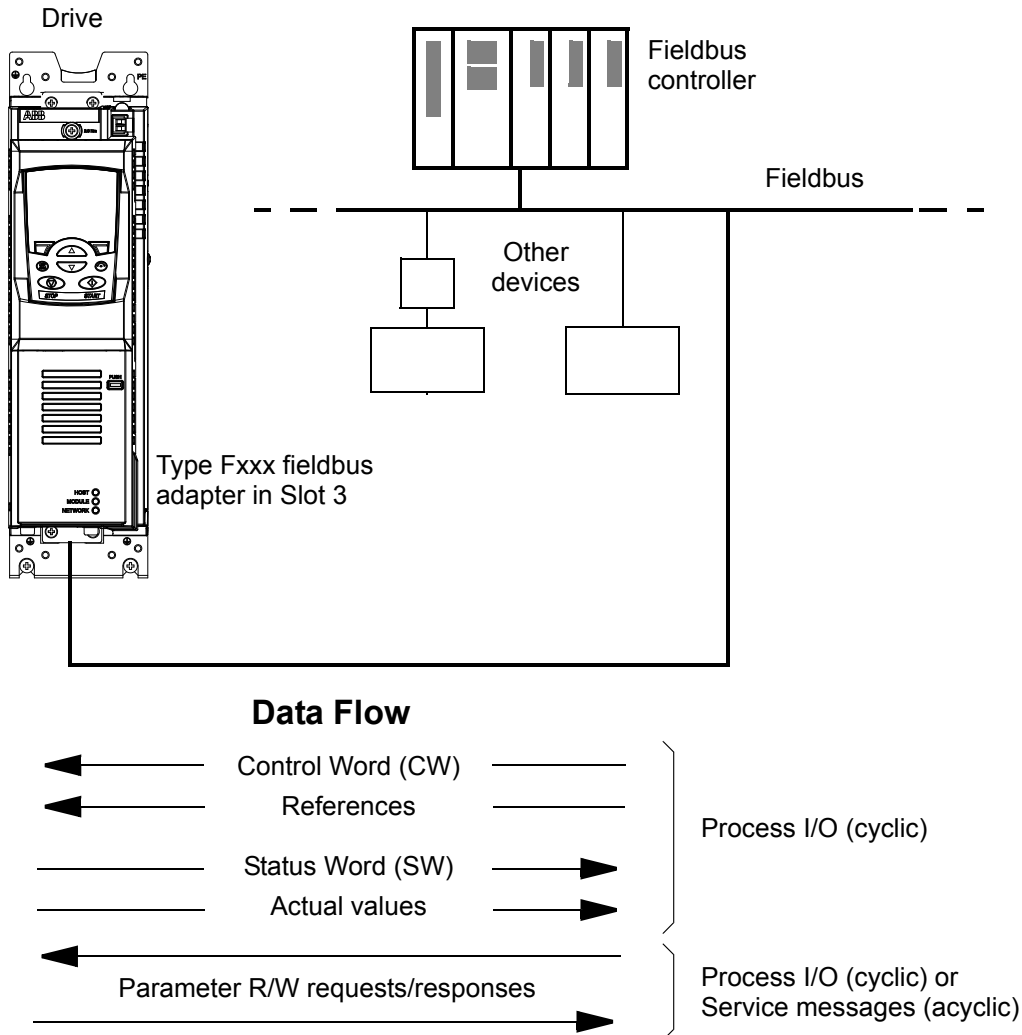
efesotomasyon.com

What this chapter contains

The chapter describes how the drive can be controlled by external devices over a communication network (fieldbus) through an optional fieldbus adapter module.

System overview

The drive can be connected to an external control system through a serial communication link using either an embedded fieldbus interface or a fieldbus adapter. The fieldbus adapter module is installed into drive Slot 3.



The drive can be set to receive all of its control information through the fieldbus interface, or the control can be distributed between the fieldbus interface and other available sources, for example, digital and analogue inputs.

Fieldbus adapters are available for various serial communication protocols, for example

- PROFIBUS DP (FPBA-xx adapter)
- CANopen (FCAN-xx adapter)
- DeviceNet (FDNA-xx adapter)
- LONWORKS[®] (FLON-xx adapter).

Setting up communication through a fieldbus adapter module

Before configuring the drive for fieldbus control, the adapter module must be mechanically and electrically installed according to the instructions given in the *User's Manual* of the appropriate fieldbus adapter module.

The communication between the drive and the fieldbus adapter module is activated by setting parameter [50.01 Fba enable](#) to *Enable*. The adapter-specific parameters must also be set. See the table below.

Parameter	Setting for fieldbus control	Function/Information
COMMUNICATION INITIALISATION AND SUPERVISION (see also page 230)		
50.01 Fba enable	(1) <i>Enable</i>	Initialises communication between drive and fieldbus adapter module.
50.02 Comm loss func	(0) <i>No</i> (1) <i>Fault</i> (2) <i>Spd ref Safe</i> (3) <i>Last speed</i>	Selects how the drive reacts upon a fieldbus communication break.
50.03 Comm loss t out	0.3...6553.5 s	Defines the time between communication break detection and the action selected with parameter 50.02 Comm loss func .
50.04 Fb ref1 modesel and 50.05 Fb ref2 modesel	(0) <i>Raw data</i> (1) <i>Torque</i> (2) <i>Speed</i>	Defines the fieldbus reference scaling. When <i>Raw data</i> is selected, see also parameters 50.06...50.11 .
50.15 Fb cw used	P.02.22	Selects the address of the fieldbus Control Word in use (02.22 FBA main cw).
ADAPTER MODULE CONFIGURATION (see also page 232)		
51.01 FBA type	–	Displays the type of the fieldbus adapter module.
51.02 FBA par2	These parameters are adapter module-specific. For more information, see the <i>User's Manual</i> of the fieldbus adapter module. Note that not all of these parameters are necessarily used.	
•••		
51.26 FBA par26		
51.27 FBA par refresh	(0) <i>Done</i> (1) <i>Refresh</i>	Validates any changed adapter module configuration parameter settings.
51.28 Par table ver	–	Displays the parameter table revision of the fieldbus adapter module mapping file stored in the memory of the drive.

Parameter	Setting for fieldbus control	Function/Information
51.29 Drive type code	–	Displays the drive type code of the fieldbus adapter module mapping file stored in the memory of the drive.
51.30 Mapping file ver	–	Displays the fieldbus adapter module mapping file revision stored in the memory of the drive.
51.31 D2FBA comm sta	–	Displays the status of the fieldbus adapter module communication.
51.32 FBA comm sw ver	–	Displays the common program revision of the adapter module.
51.33 FBA appl sw ver	–	Displays the application program revision of the adapter module.
Note: In the <i>User's Manual</i> of the fieldbus adapter module, the parameter group number is 1 or A for parameters 51.01 ... 51.26 .		
TRANSMITTED DATA SELECTION (see also page 233)		
52.01 FBA data in1 ... 52.12 FBA data in12	4...6 14...16 101...9999	Defines the data transmitted from drive to fieldbus controller. Note: If the selected data is 32 bits long, two parameters are reserved for the transmission.
53.01 FBA data out1 ... 53.12 FBA data out12	1...3 11...13 1001...9999	Defines the data transmitted from fieldbus controller to drive. Note: If the selected data is 32 bits long, two parameters are reserved for the transmission.
Note: In the <i>User's Manual</i> of the fieldbus adapter module, the parameter group number is 2 or B for parameters 52.01 ... 52.12 and 3 or C for parameters 53.01 ... 53.12 .		

After the module configuration parameters have been set, the drive control parameters (see section [Setting the drive control parameters](#) below) must be checked and adjusted when necessary.

The new settings will take effect when the drive is powered up the next time, or when parameter [51.27 FBA par refresh](#) is activated.

Setting the drive control parameters

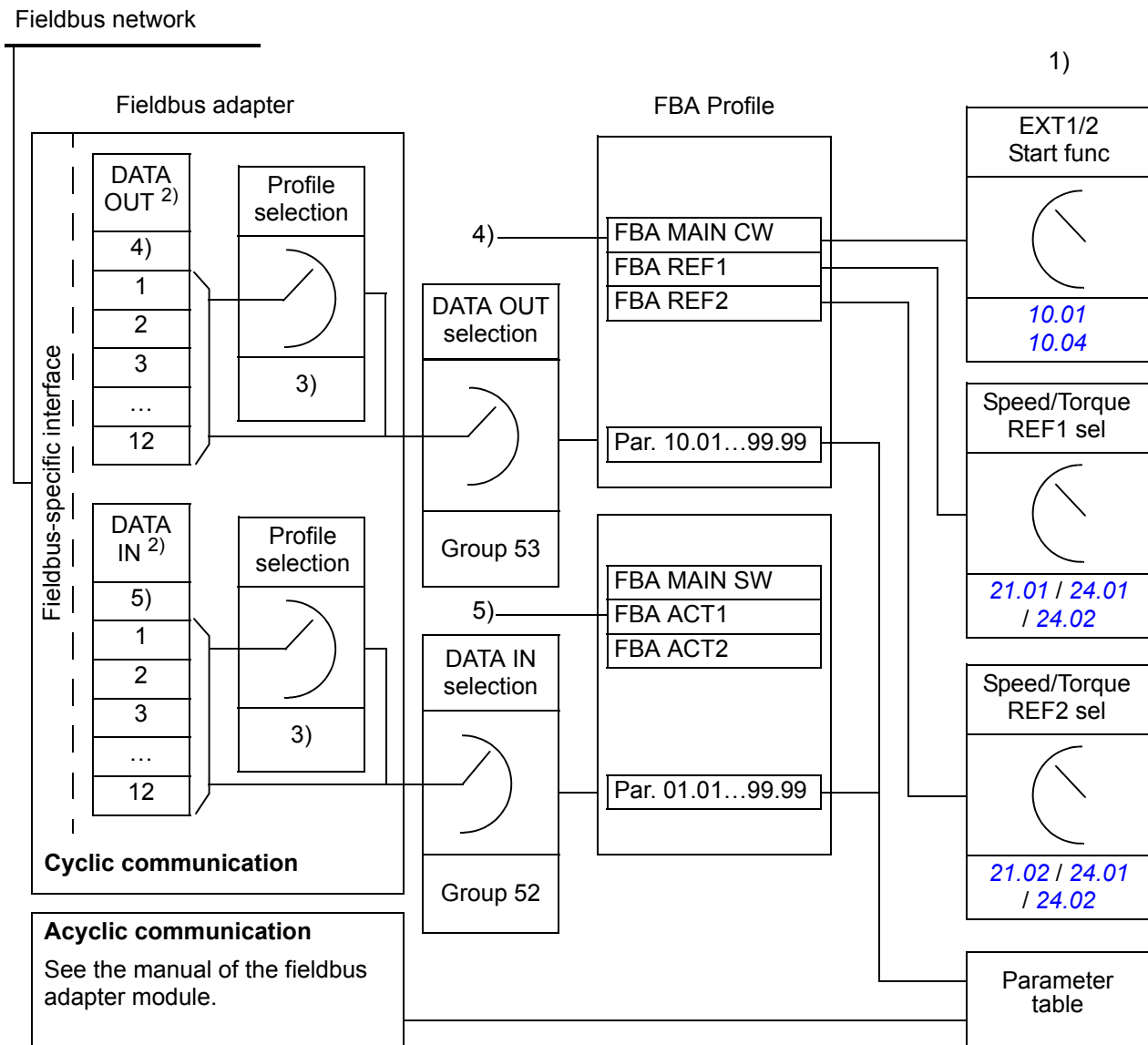
The **Setting for fieldbus control** column gives the value to use when the fieldbus interface is the desired source or destination for that particular signal. The **Function/Information** column gives a description of the parameter.

Parameter	Setting for fieldbus control	Function/Information
CONTROL COMMAND SOURCE SELECTION		
<i>10.01 Ext1 start func</i>	(3) <i>FB</i>	Selects fieldbus as the source for the start and stop commands when EXT1 is selected as the active control location.
<i>10.04 Ext2 start func</i>	(3) <i>FB</i>	Selects fieldbus as the source for the start and stop commands when EXT2 is selected as the active control location.
<i>21.01 Speed ref1 sel</i>	(3) <i>FBA ref1</i> (4) <i>FBA ref2</i>	Fieldbus reference REF1 or REF2 is used as speed reference 1.
<i>21.02 Speed ref2 sel</i>	(3) <i>FBA ref1</i> (4) <i>FBA ref2</i>	Fieldbus reference REF1 or REF2 is used as speed reference 2.
<i>24.01 Torq ref1 sel</i>	(3) <i>FBA ref1</i> (4) <i>FBA ref2</i>	Fieldbus reference REF1 or REF2 is used as torque reference 1.
<i>24.02 Torq ref add sel</i>	(3) <i>FBA ref1</i> (4) <i>FBA ref2</i>	Fieldbus reference REF1 or REF2 is used as torque reference addition.
SYSTEM CONTROL INPUTS		
<i>16.07 Param save</i>	(0) <i>Done</i> (1) <i>Save</i>	Saves parameter value changes (including those made through fieldbus control) to permanent memory.

Basics of the fieldbus adapter interface

The cyclic communication between a fieldbus system and the drive consists of 16/32-bit input and output data words. The drive supports at the maximum the use of 12 data words (16 bits) in each direction.

Data transmitted from the drive to the fieldbus controller is defined by parameters [52.01 FBA data in1](#) ... [52.12 FBA data in12](#). The data transmitted from the fieldbus controller to the drive is defined by parameters [53.01 FBA data out1](#) ... [53.12 FBA data out12](#).



- 1) See also other parameters which can be controlled by the fieldbus.
- 2) The maximum number of used data words is protocol-dependent.
- 3) Profile/instance selection parameters. Fieldbus module specific parameters. For more information, see the *User's Manual* of the appropriate fieldbus adapter module.
- 4) With DeviceNet, the control part is transmitted directly.
- 5) With DeviceNet, the actual value part is transmitted directly.

■ Control Word and Status Word

The Control Word (CW) is the principal means of controlling the drive from a fieldbus system. The Control Word is sent by the fieldbus controller to the drive. The drive switches between its states according to the bit-coded instructions of the Control Word.

The Status word (SW) is a word containing status information, sent by the drive to the fieldbus controller.

■ Actual values

Actual values (ACT) are 16/32-bit words containing information on selected operations of the drive.

FBA communication profile

The FBA communication profile is a state machine model which describes the general states and state transitions of the drive. The [State diagram](#) on page 339 presents the most important states (including the FBA profile state names). The FBA Control Word (parameter [02.22 FBA main cw](#) – see page 108) commands the transitions between these states and the FBA Status Word (parameter [02.24 FBA main sw](#) – see page 109) indicates the status of the drive.

Fieldbus adapter module profile (selected by adapter module parameter) defines how the Control Word and Status Word are transmitted in a system which consists of fieldbus controller, fieldbus adapter module and drive. With transparent modes, Control Word and Status Word are transmitted without any conversion between the fieldbus controller and the drive. With other profiles (e.g. PROFIdrive for FPBA-01, AC/DC drive for FDNA-01, DS-402 for FCAN-01 and ABB Drives profile for all fieldbus adapter modules), the fieldbus adapter module converts the fieldbus-specific Control word to the FBA communication profile and Status Word from FBA communication profile to the fieldbus-specific Status Word.

For descriptions of other profiles, see the *User's Manual* of the appropriate fieldbus adapter module.

■ Fieldbus references

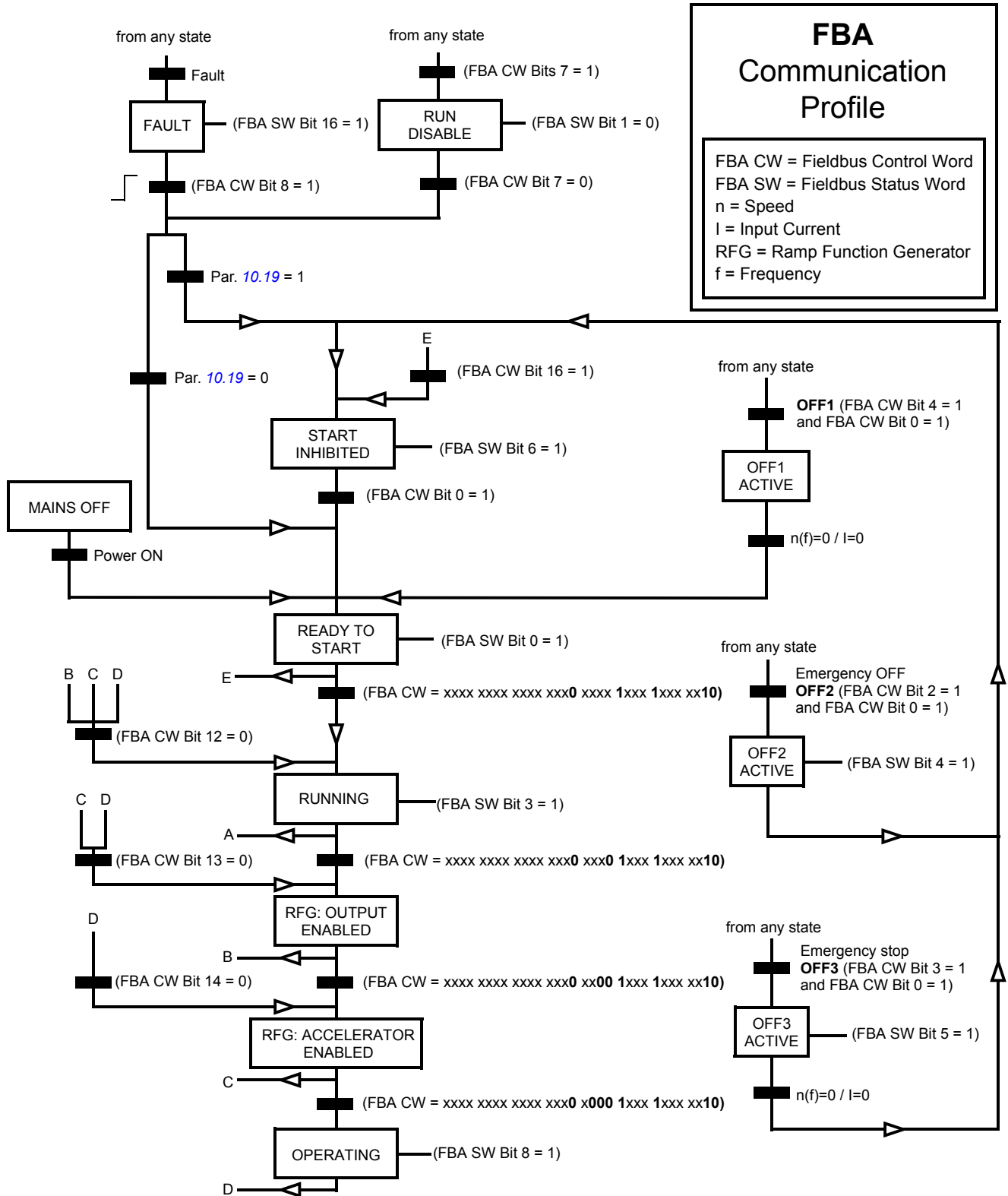
References (FBA REF) are 16/32-bit signed integers. A negative reference (indicating reversed direction of rotation) is formed by calculating the two's complement from the corresponding positive reference value. The contents of each reference word can be used as torque or speed reference.

When torque or speed reference scaling is selected (by parameter [50.04 Fb ref1 modesel](#) / [50.05 Fb ref2 modesel](#)), the fieldbus references are 32-bit integers. The value consists of a 16-bit integer value and a 16-bit fractional value. The speed/torque reference scaling is as follows:

Reference	Scaling	Notes
Speed reference	FBA REF / 65536 (value in rpm)	Final reference is limited by parameters 20.01 Maximum speed , 20.02 Minimum speed and 21.09 SpeedRef min abs .
Torque reference	FBA REF / 65536 (value in %)	Final reference is limited by torque limit parameters 20.06...20.10 .

■ State diagram

The following presents the state diagram for the FBA communication profile. For other profiles, see the *User's Manual* of the appropriate fieldbus adapter module.





Drive-to-drive link

What this chapter contains

efesotomasyon.com

The chapter describes the communication on the drive-to-drive link.

General

The drive-to-drive link is a daisy-chained RS-485 transmission line, constructed by connecting the XD2D terminal blocks of the JCU control units of several drives. It is also possible to use an FMBA Modbus adapter module installed into an option slot on the JCU. The firmware supports up to 63 nodes on the link.

The link has one master drive; the rest of the drives are followers. By default, the master broadcasts control commands as well as speed and torque references for all followers. The master can send 8 messages per millisecond at 100/150-microsecond intervals. Sending one message takes approximately 15 microseconds, which results in a theoretical link capacity of roughly 6 messages per 100 microseconds.

Multicasting the control data and reference 1 to a pre-defined group of drives is possible, as is chained multicast messaging. Reference 2 is always broadcast by the master to all followers. See parameters [57.11](#)...[57.14](#).

Note: The drive-to-drive link can be used only if the embedded fieldbus interface is disabled (see parameter [58.01 Protocol ena sel](#)).

■ Wiring

See the *Hardware Manual* of the drive.

Datasets

Drive-to-drive communication uses DDCS (Distributed Drives Communication System) messages and dataset tables for data transfer. Each drive has a dataset table of 256 datasets, numbered 0...255. Each dataset contains 48 data bits.

By default, datasets 0...15 and 200...255 are reserved for the drive firmware; datasets 16...199 are available for the user application program.

The contents of the two firmware communication datasets can be configured freely with pointer parameters and/or application programming with the DriveSPC tool. The 16-bit control word and 32-bit drive-to-drive reference 1 are transmitted from one dataset on a 500-microsecond (by default) time level; drive-to-drive reference 2 (32 bits) is transmitted from the other dataset on a 2-millisecond (by default) time level. The followers can be configured to use the drive-to-drive commands and references with the following parameters:

Control data	Parameter	Setting for drive-to-drive communication
Start/Stop commands	10.01 Ext1 start func 10.04 Ext2 start func	<i>D2D</i>
Torque limits	20.09 Maximum torque2 20.10 Minimum torque2	<i>D2D ref1</i> or <i>D2D ref2</i>
Speed reference	21.01 Speed ref1 sel 21.02 Speed ref2 sel 23.08 Speed additive	<i>D2D ref1</i> or <i>D2D ref2</i>
Torque reference	24.01 Torq ref1 sel 24.02 Torq ref add sel	<i>D2D ref1</i> or <i>D2D ref2</i>
PID setpoint and feedback	27.01 PID setpoint sel 27.03 PID fbk1 src 27.04 PID fbk2 src	<i>D2D ref1</i> or <i>D2D ref2</i>
Mechanical brake open torque	42.09 Open torq src	<i>D2D ref1</i> or <i>D2D ref2</i>

The communication status of the followers can be supervised by a periodic supervision message from the master to the individual followers (see parameters [57.04 Follower mask 1](#) and [57.05 Follower mask 2](#)).

Drive-to-drive function blocks can be used in the DriveSPC tool to enable additional communication methods (such as follower-to-follower messaging) and to modify the use of datasets between the drives. See separate document *Application guide: Application programming for ACS850 drives* (3AUA0000078664 [English]).

Types of messaging

Each drive on the link has a unique node address allowing point-to-point communication between two drives. The node address 0 is automatically assigned to the master drive; on other drives, the node address is defined by parameter [57.03 Node address](#).

Multicast addressing is supported, allowing the composition of groups of drives. Data sent to a multicast address is received by all drives that have that address. A multicast group can consist of 1...62 drives.

In broadcast messaging, data can be sent to all drives (actually, all followers) on the link.

Both master-to-follower(s) and follower-to-follower(s) communication is supported. A follower can send one message to another follower (or a group of followers) after receiving a token message from the master.

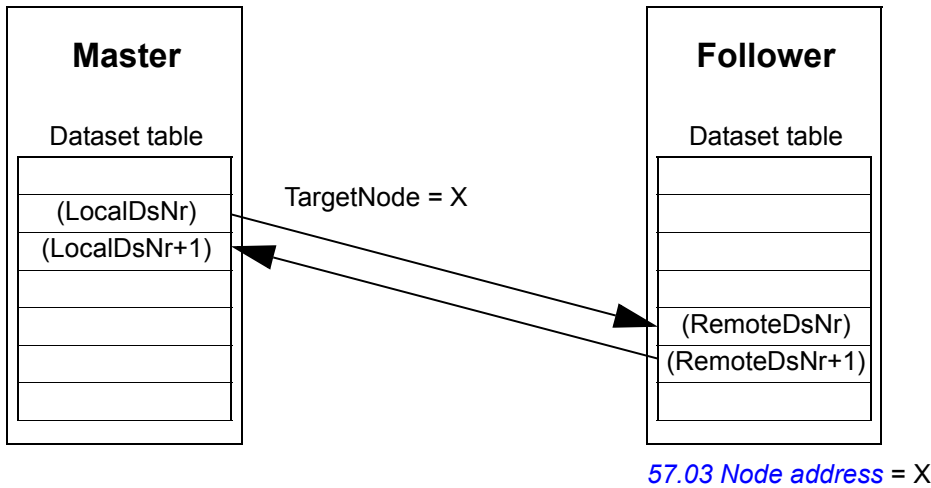
Type of messaging		Note
Point-to-point	Master point-to-point	Supported only at master
	Read remote	Supported only at master
	Follower point-to-point	Supported only at followers
Standard multicast		For both master and followers
Broadcast		For both master and followers
Token message for follower-to-follower communication		–
Chained multicast		Supported only for drive-to-drive reference 1 and control word

■ Master point-to-point messaging

In this type of messaging, the master sends one dataset (LocalDsNr) from its own dataset table to the follower's. TargetNode stands for the node address of the follower; RemoteDsNr specifies the target dataset number.

The follower responds by returning the contents of the next dataset. The response is stored into dataset LocalDsNr+1 in the master.

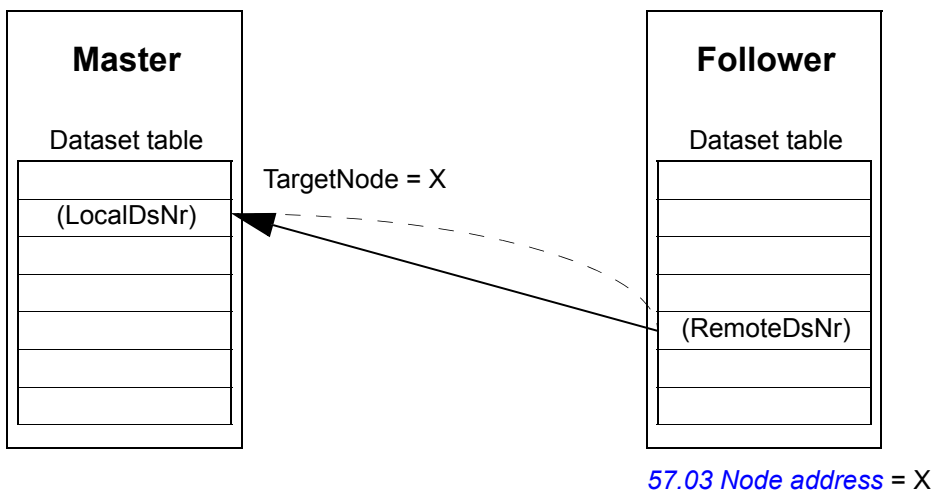
Note: Master point-to-point messaging is only supported at the master because the response is always sent to node address 0 (the master).



■ Read remote messaging

The master can read a dataset (RemoteDsNr) from a follower specified by TargetNode. The follower returns the contents of the requested dataset to the master. The response is stored at dataset LocalDsNr in the master.

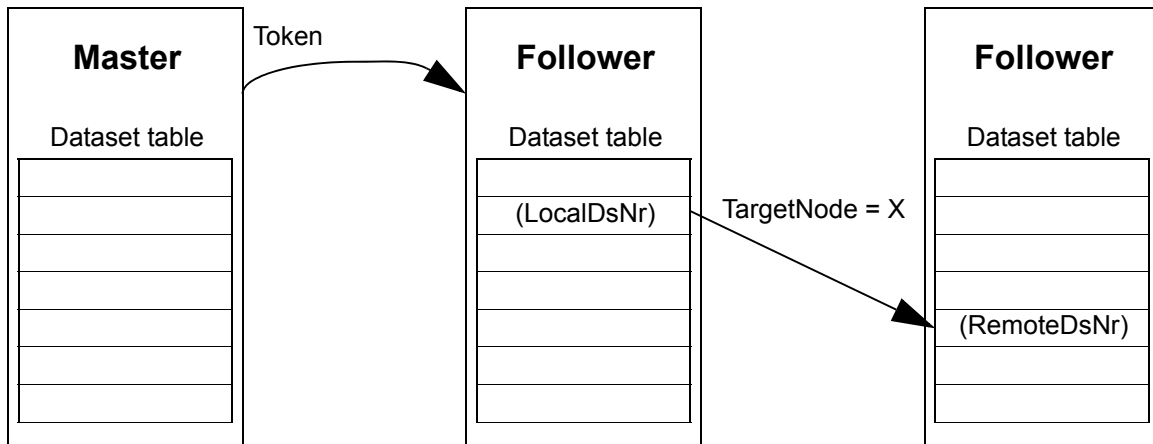
Note: Read remote messaging is only supported at the master because the response is always sent to node address 0 (the master).



■ Follower point-to-point messaging

This type of messaging is for point-to-point communication between followers. After receiving a token from the master, a follower can send one dataset to another follower with a follower point-to-point message. The target drive is specified using the node address.

Note: The data is not sent to the master.



57.03 Node address = X

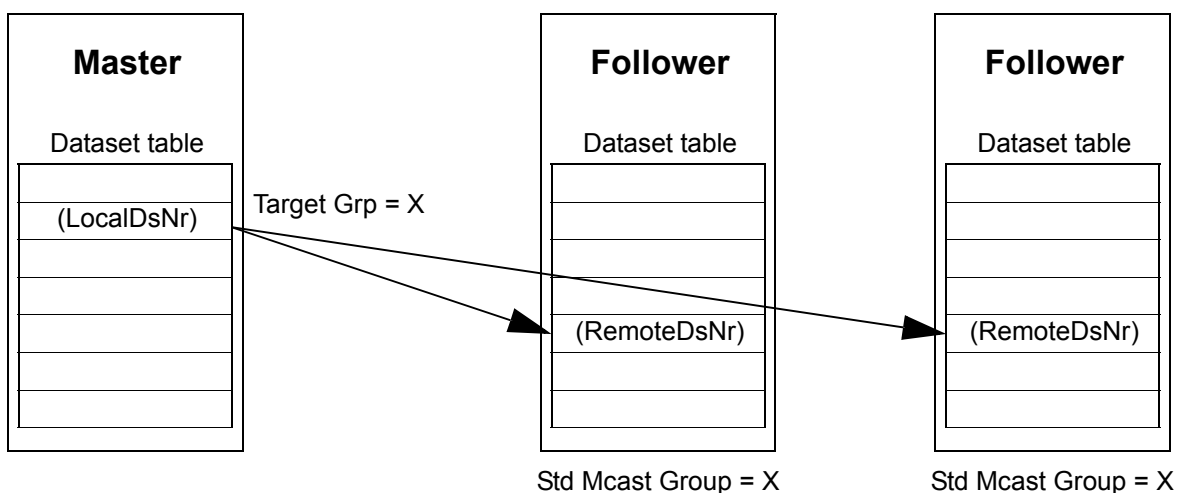
■ Standard multicast messaging

In standard multicast messaging, one dataset can be sent to a group of drives having the same standard multicast group address. The target group is defined by the *D2D_Conf* standard function block; see separate document *Application programming for ACS850 drives* (3AUA0000078664 [English]).

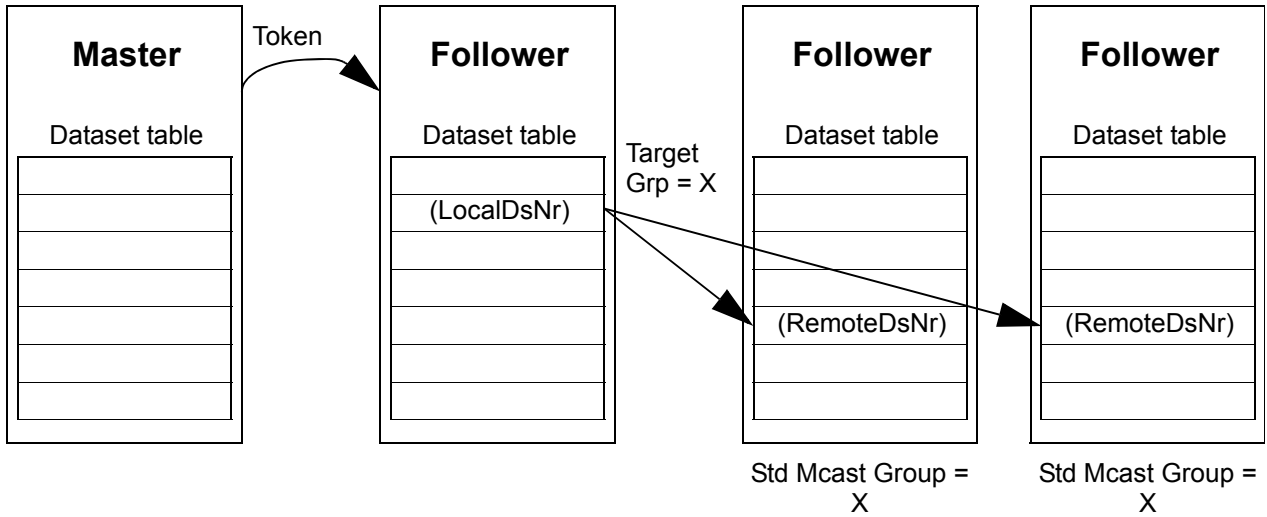
The sending drive can either be the master, or a follower after receiving a token from the master.

Note: The master does not receive the sent data even if it is a member of the target multicast group.

Master-to-follower(s) multicasting



Follower-to-follower(s) multicasting



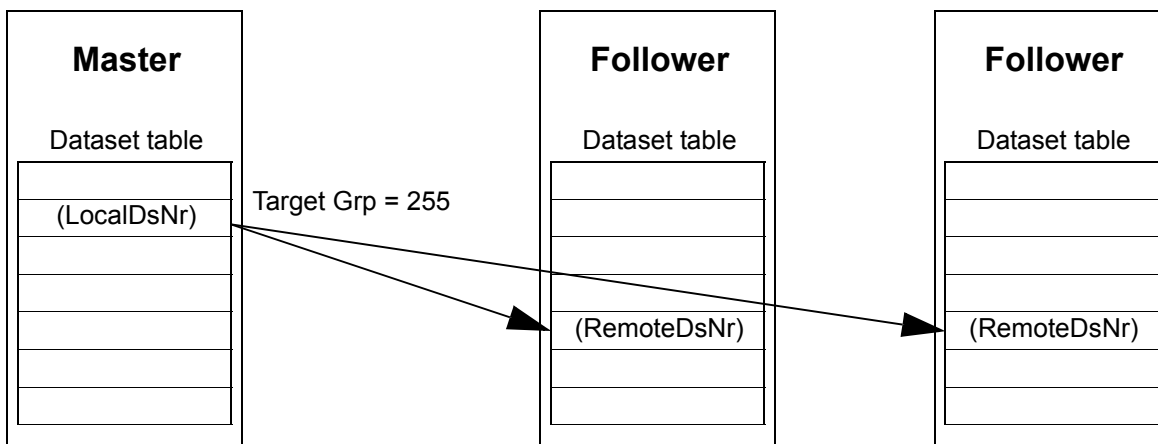
■ **Broadcast messaging**

In broadcasting, the master sends one dataset to all followers, or a follower sends one dataset to all other followers (after receiving a token from the master).

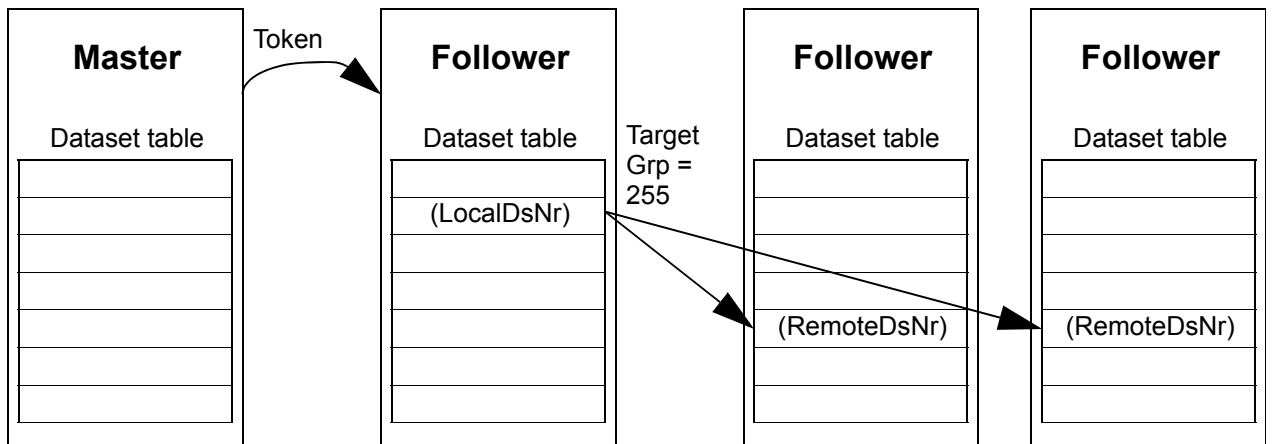
The target (Target Grp) is automatically set to 255 denoting all followers.

Note: The master does not receive any data broadcast by the followers.

Master-to-follower(s) broadcasting



Follower-to-follower(s) broadcasting



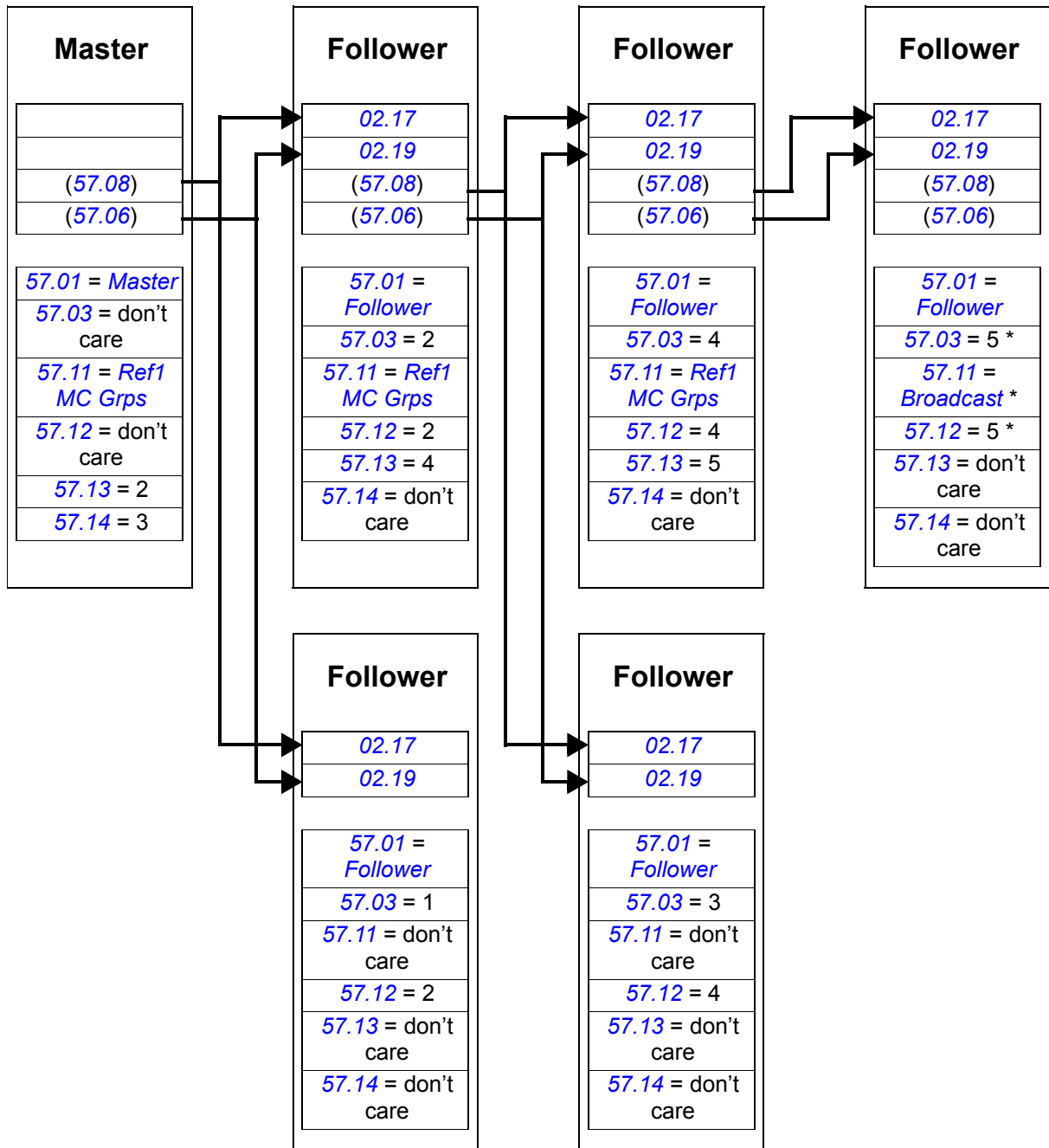
■ Chained multicast messaging

Chained multicasting is supported for drive-to-drive reference 1 and Control word by the firmware.

The message chain is always started by the master. The target group is defined by parameter [57.13 Next ref1 mc grp](#). The message is received by all followers that have parameter [57.12 Ref1 mc group](#) set to the same value as parameter [57.13 Next ref1 mc grp](#) in the master.

If a follower has parameters [57.03 Node address](#) and [57.12 Ref1 mc group](#) set to the same value, it becomes a submaster. Immediately after a submaster receives the multicast message, it sends its own message to the next multicast group defined by parameter [57.13 Next ref1 mc grp](#).

The duration of the entire message chain is approximately 15 microseconds multiplied by the number of links in the chain (defined by parameter [57.14 Nr ref1 mc grps](#) in the master).



* Acknowledgement from last follower to master can be prevented by setting parameter **57.11 Ref1 msg type** to **Broadcast** (required because parameters **57.03 Node address** and **57.12 Ref1 mc group** are set to the same value). Alternatively, node/group addresses (parameters **57.03 Node address** and **57.12 Ref1 mc group**) could be set to non-equal values.



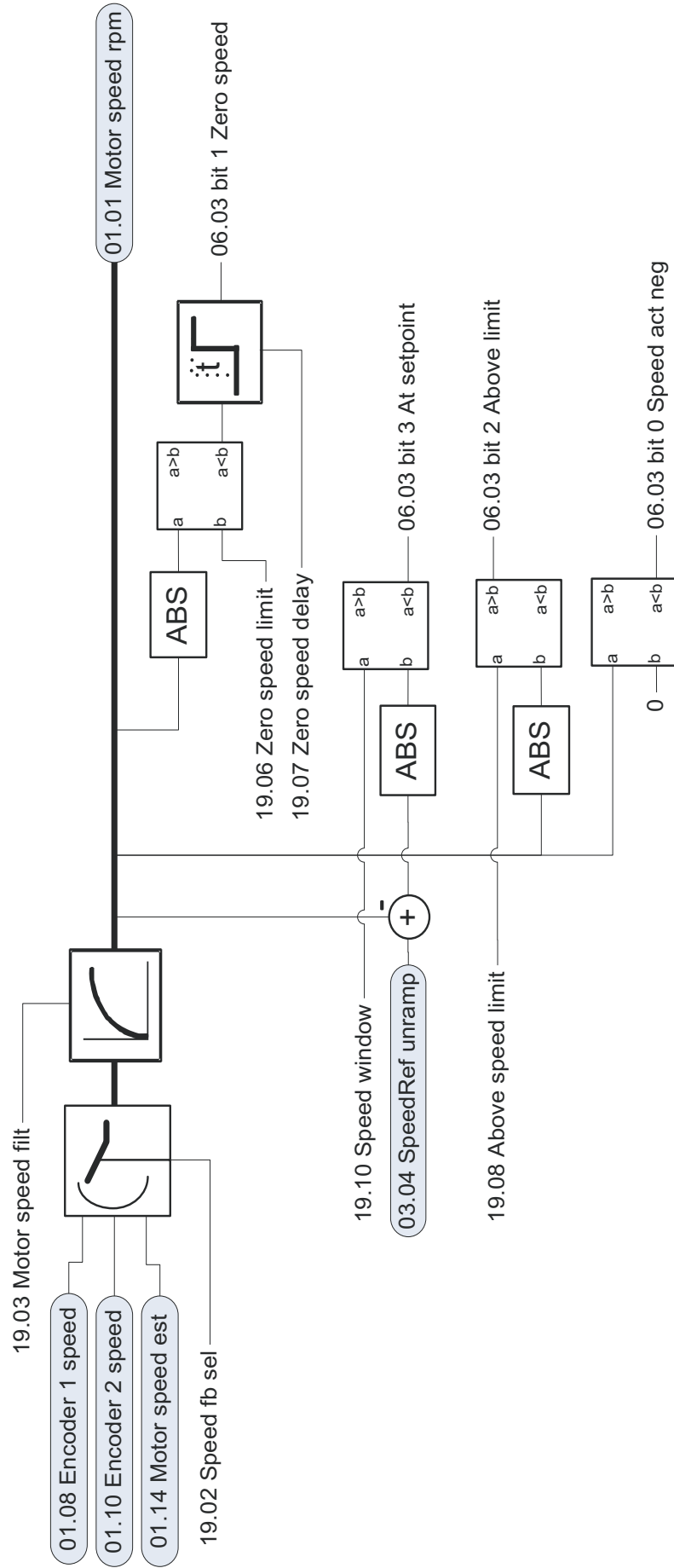
Control chain and drive logic diagrams

efesotomasyon.com

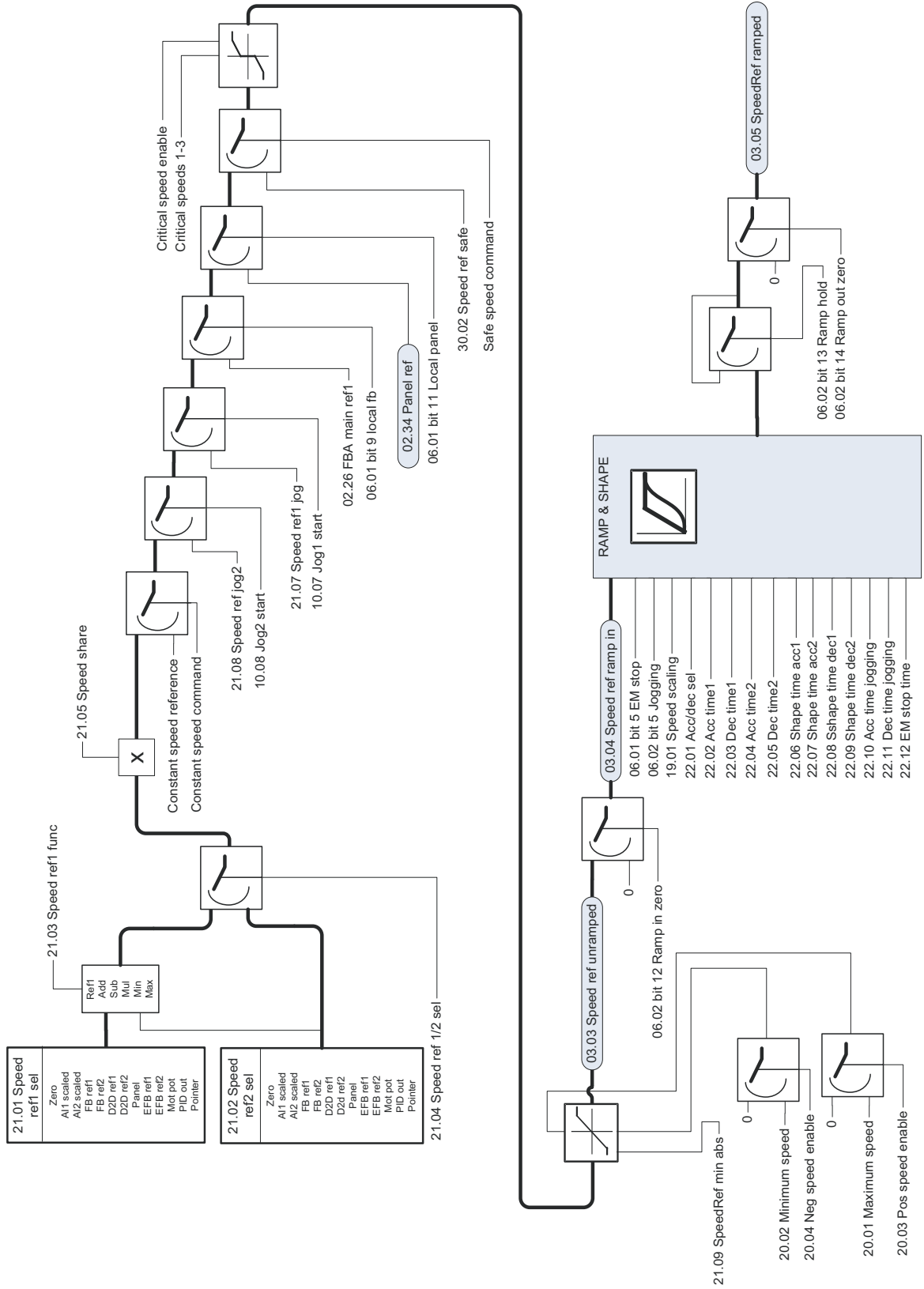
What this chapter contains

The chapter presents the drive control chain and logic.

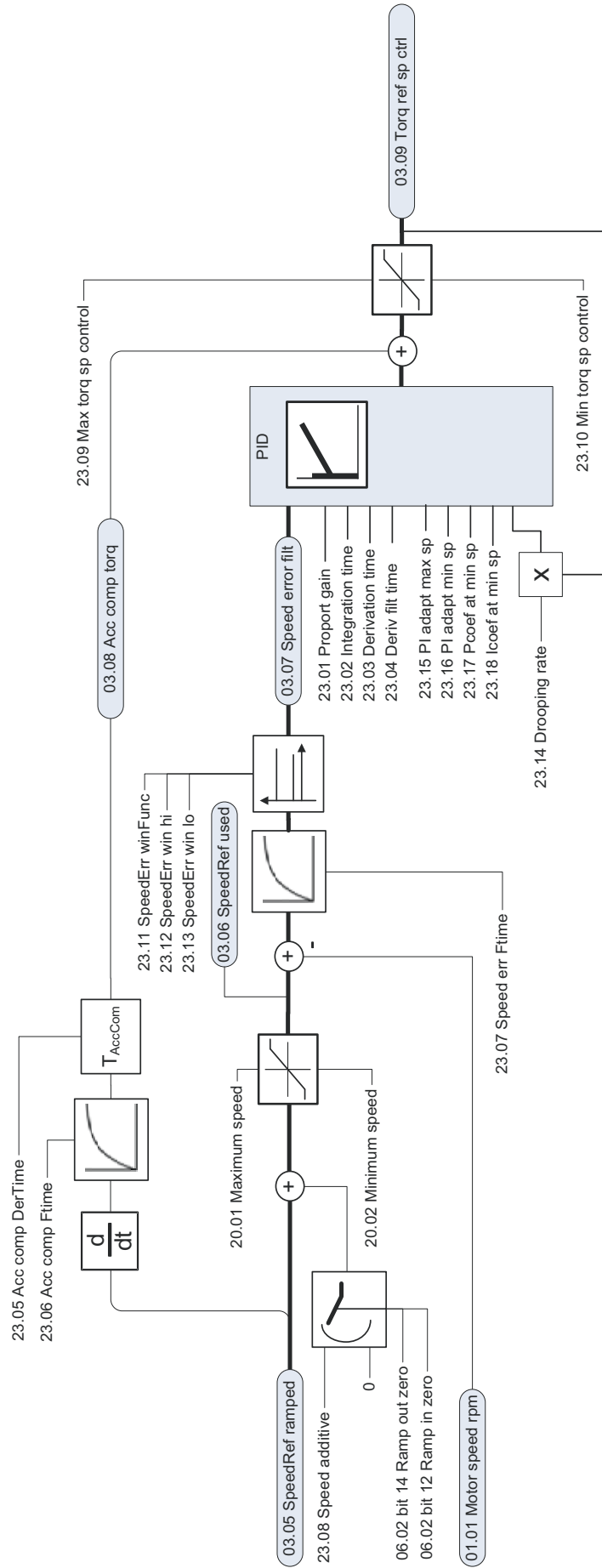
Speed feedback



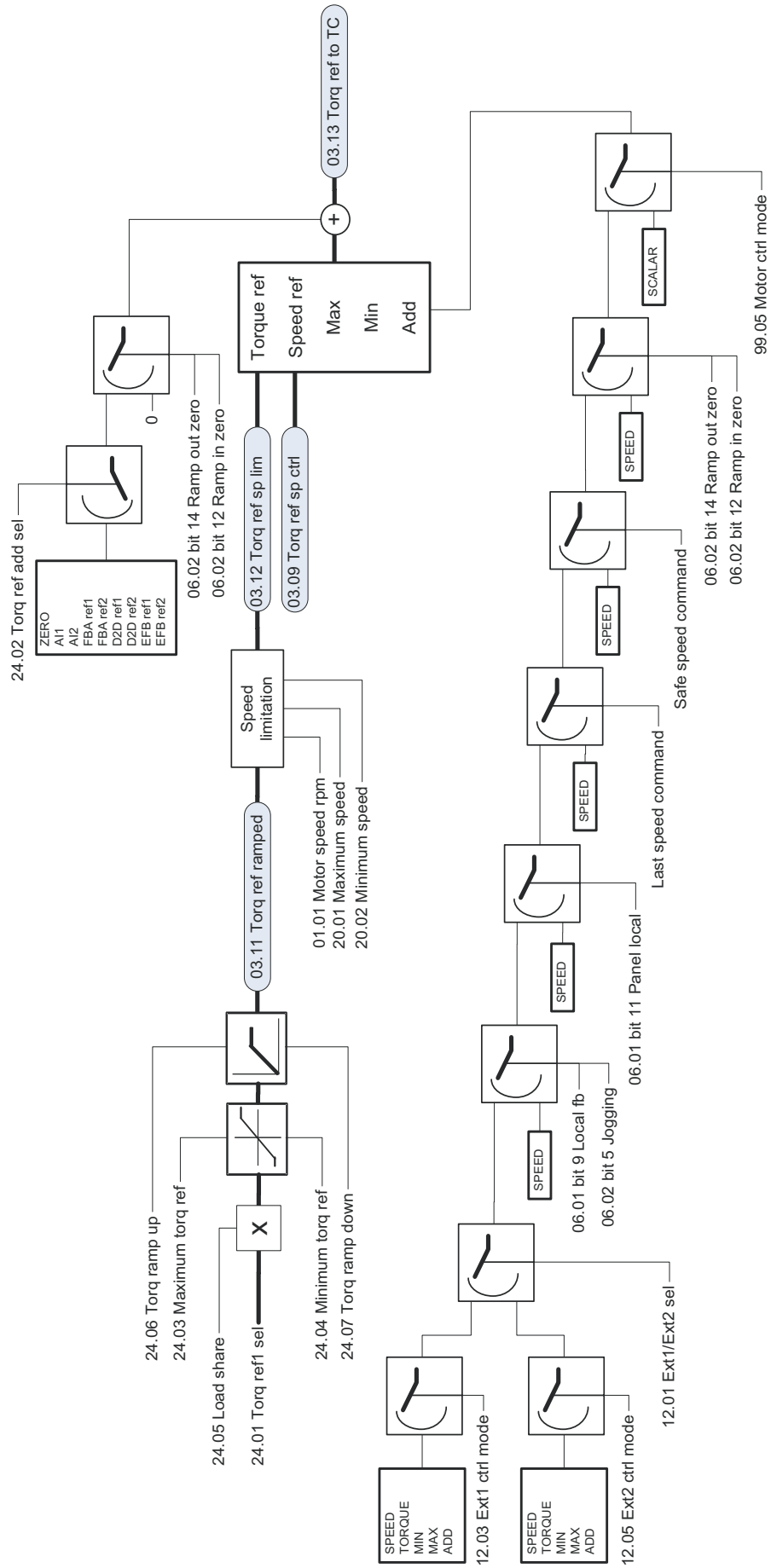
Speed reference modification and ramping



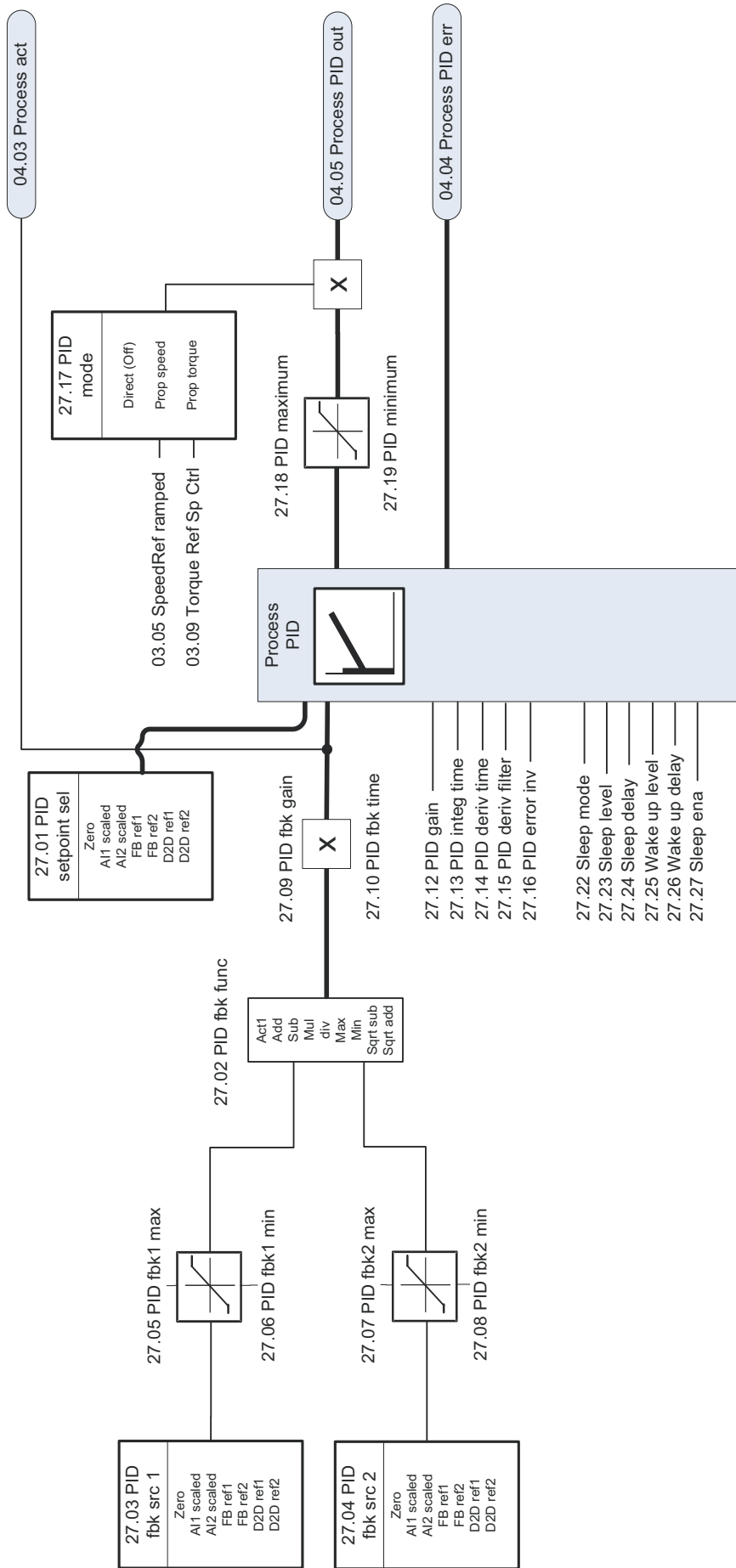
Speed error handling



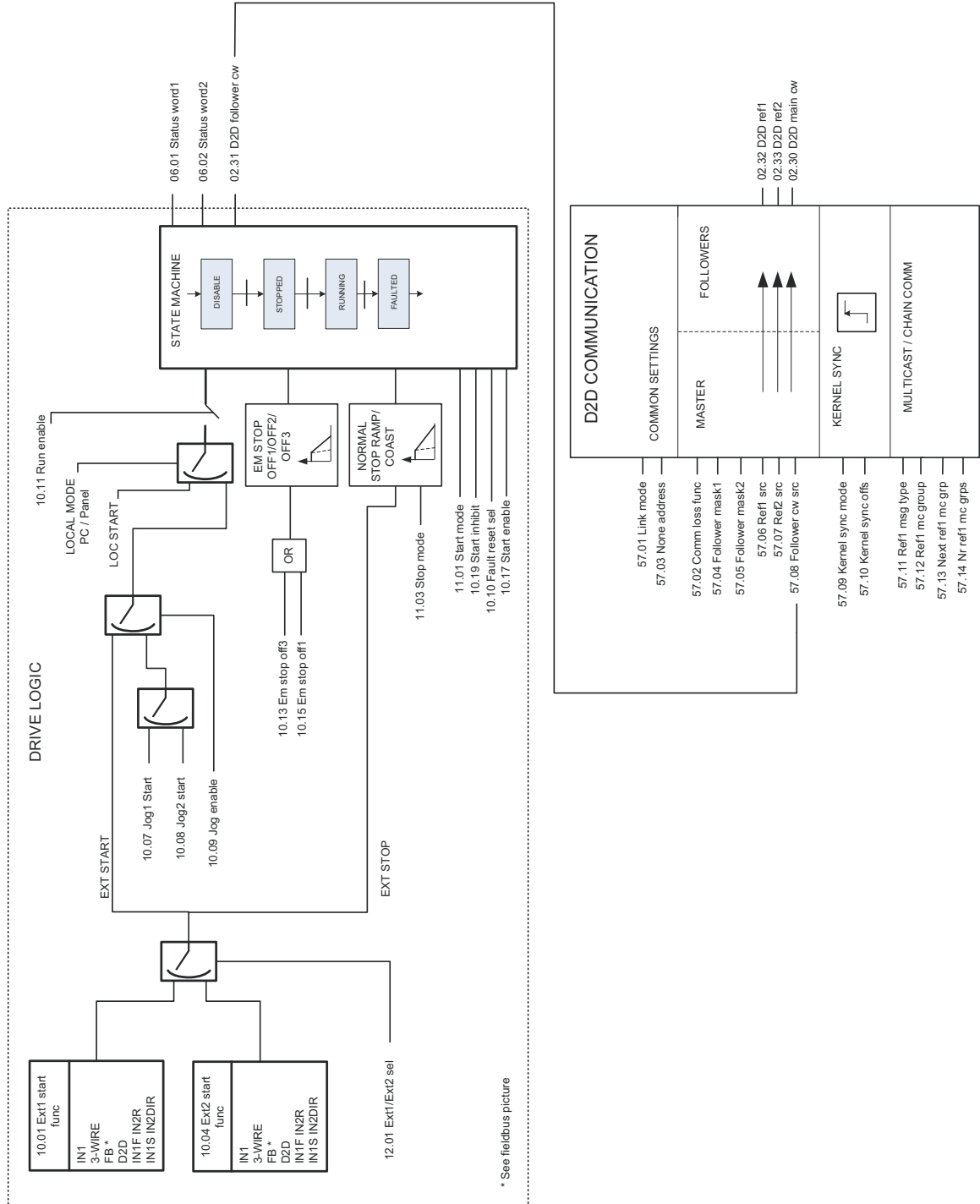
Torque reference modification, operating mode selection



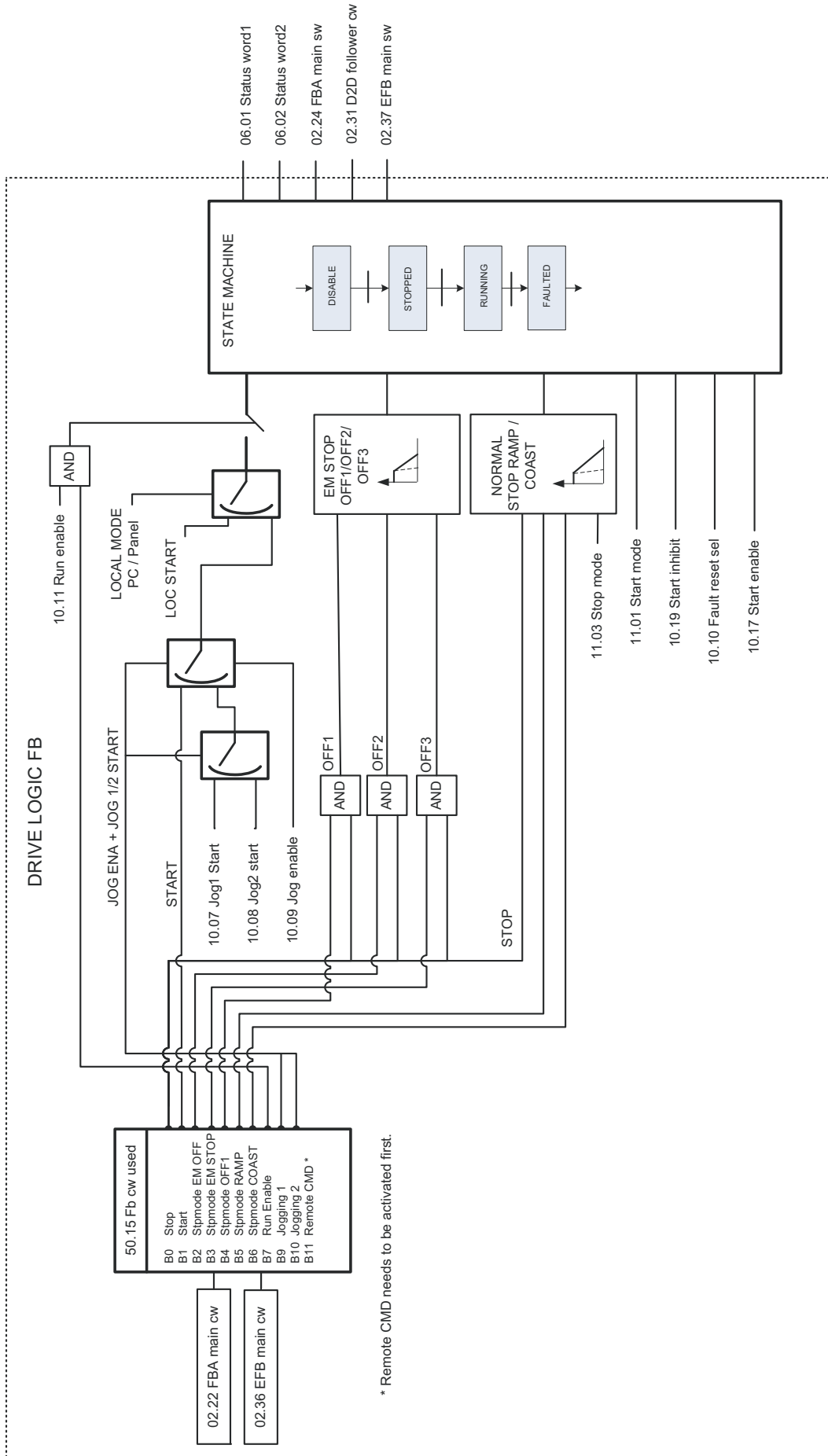
Process PID



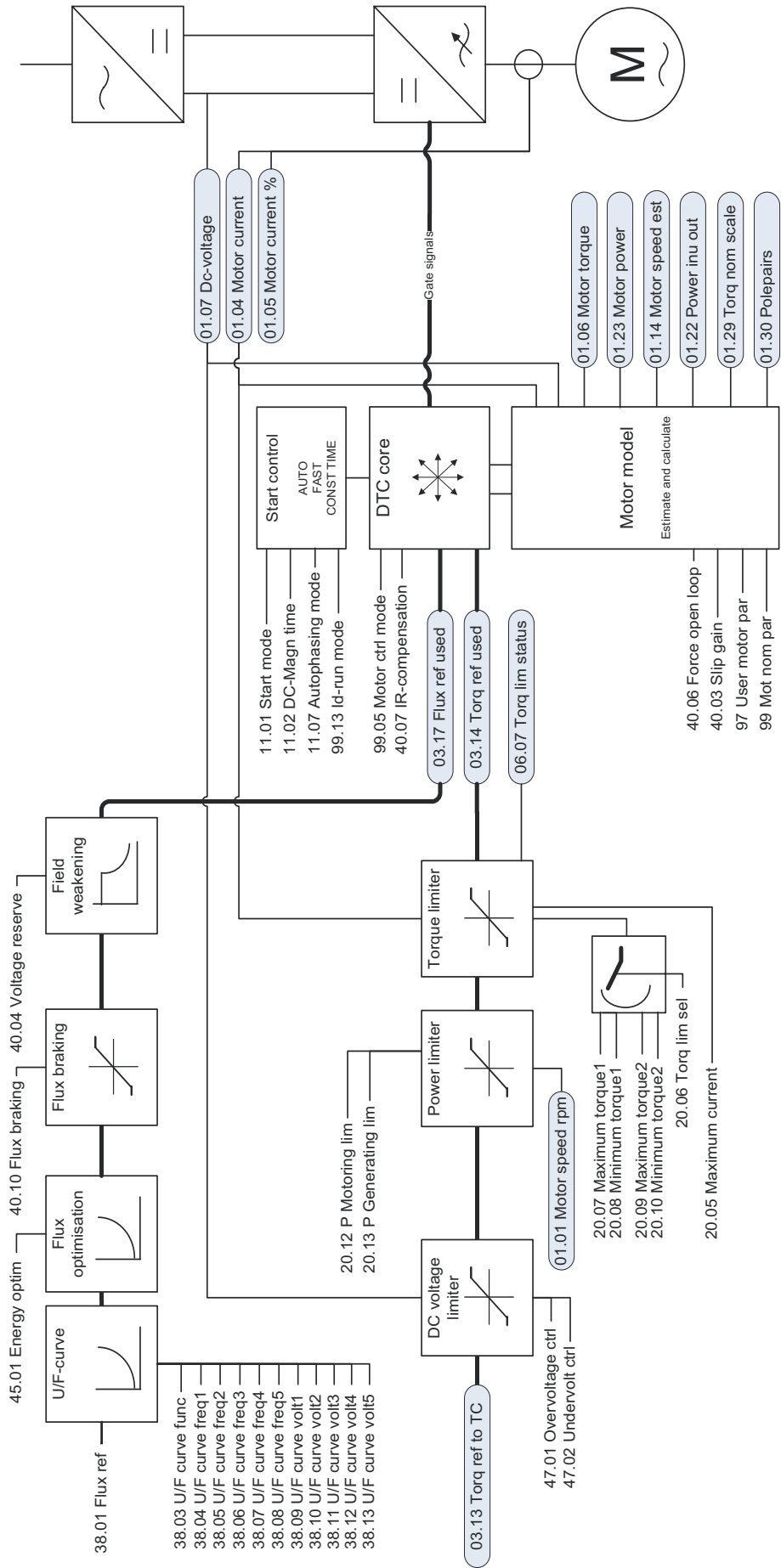
Drive logic 1



Drive logic 2 (Fieldbus interface)



Direct torque control



Further information efesotomasyon.com

Product and service inquiries

Address any inquiries about the product to your local ABB representative, quoting the type designation and serial number of the unit in question. A listing of ABB sales, support and service contacts can be found by navigating to www.abb.com/drives and selecting *Sales, Support and Service network*.

Product training

For information on ABB product training, navigate to www.abb.com/drives and select *Training courses*.

Providing feedback on ABB Drives manuals

Your comments on our manuals are welcome. Go to www.abb.com/drives and select *Document Library – Manuals feedback form (LV AC drives)*.

Document library on the Internet

You can find manuals and other product documents in PDF format on the Internet. Go to www.abb.com/drives and select *Document Library*. You can browse the library or enter selection criteria, for example a document code, in the search field.



3AUA0000045497 Rev E / EN
EFFECTIVE: 2010-10-12

ABB Oy

Drives
P.O. Box 184
FI-00381 HELSINKI
FINLAND
Telephone +358 10 22 11
Fax +358 10 22 22681
www.abb.com/drives

ABB Inc.

Automation Technologies
Drives & Motors
16250 West Glendale Drive
New Berlin, WI 53151 USA
Telephone 262 785-3200
1-800-HELP-365
Fax 262 780-5135
www.abb.com/drives

ABB Beijing Drive Systems Co. Ltd.

No. 1, Block D, A-10 Jiuxianqiao Beilu
Chaoyang District
Beijing, P.R. China, 100015
Telephone +86 10 5821 7788
Fax +86 10 5821 7618
www.abb.com/drives

ABB industrial drives

Application guide

Application programming for ACS850 drives

List of related manuals

Drive hardware manuals and guides

	Code (English)	
<i>ACS850-04 (Frames A to D) hardware manual</i>	3AUA0000045496	1)
<i>ACS850-04 (Frames E0 and E) hardware manual</i>	3AUA0000045487	1)
<i>ACS850-04 (Frame G) hardware manual</i>	3AUA0000026234	1)

Drive firmware manuals and guides

<i>ACS850 standard control program quick start-up guide</i>	3AUA0000045498	2)
<i>ACS850 standard control program firmware manual</i>	3AUA0000045497	3)

Drive PC tools manuals

<i>DriveSPC user manual</i>	3AFE68836590
<i>DriveStudio user manual</i>	3AFE68749026

- 1) Delivered as a printed copy with the drive if the order includes printed manuals.
- 2) Delivered as a printed copy with the control program.
- 3) Delivered as a printed copy with the control program if the order includes printed manuals.

All manuals are available in PDF format on the Internet. See section [Document library on the Internet](#) on the inside of the back cover.

Application guide

Application programming for
ACS850 drives

Table of contents



Table of contents

List of related manuals	2
-------------------------------	---

1. About the manual

What this chapter contains	7
Compatibility	7
Safety instructions	7
Reader	7
Purpose of the manual	8
Contents of the manual	8

2. Drive programming

What this chapter contains	9
General about drive programming	9
Application programming	10
Function blocks	10
Firmware function blocks	10
Standard function blocks	11
User parameters	11
Application events	11
Program execution	11
Application program licensing and protection	11
Operation modes	12
Off-line	12
On-line	12

3. Firmware function blocks

What this chapter contains	13
----------------------------------	----

4. Standard function blocks

What this chapter contains	17
Terms	17
Alphabetical index	18
Arithmetic	19
Bitstring	23
Bitwise	27
Communication	31
Comparison	36
Conversion	38
Counters	46
Edge & bistable	52
Extensions	55
Feedback & algorithms	64
Filters	74
Parameters	77
Program structure	80



Selection	83
Switch & Demux	85
Timers	87

5. Examples of using standard function blocks

What this chapter contains	91
Start/stop using analog input	91
Relay output and digital input/output control	93
Drive-to-drive communication	95
Example of master point-to-point messaging	95
Example of read remote messaging	95
Example of releasing tokens for follower-to-follower communication	96
Example of follower point-to-point messaging	96
Example of standard multicast messaging	97
Example of broadcast messaging	97

Further information

Product and service inquiries	99
Product training	99
Providing feedback on ABB Drives manuals	99
Document library on the Internet	99





About the manual

What this chapter contains

This chapter describes the contents of this manual. It also contains information on the compatibility, safety, intended audience and the purpose of the manual.

Compatibility

This manual is compatible with ACS850 drives with the Standard control program.

Safety instructions

Follow all safety instructions delivered with the drive.

- Read the **complete safety instructions** before you install, commission, or use the drive. The complete safety instructions are given at the beginning of the drive *Hardware manual*.
- Read the **software function specific warnings and notes** before changing the default settings of the function. For each function, the warnings and notes are given in this manual in the section describing the related user-adjustable parameters.

Reader

The reader of the manual is expected to know the standard electrical wiring practices, electronic components, and electrical schematic symbols.

Purpose of the manual

The purpose of this manual is to provide the reader with the information needed in designing application programs for ACS850 drives using the DriveSPC PC tool.

The manual is intended to be used together with the drive *Firmware manual*, which contains the basic information on the drive parameters.

Contents of the manual

The manual consists of the following chapters:

- [Drive programming](#) introduces drive programming and describes application programming using the DriveSPC PC tool.
 - [Firmware function blocks](#) presents the firmware function blocks available.
 - [Standard function blocks](#) presents the standards function blocks available.
 - [Examples of using standard function blocks](#) contains examples of using standard function blocks in application programming.
-

2

Drive programming

What this chapter contains

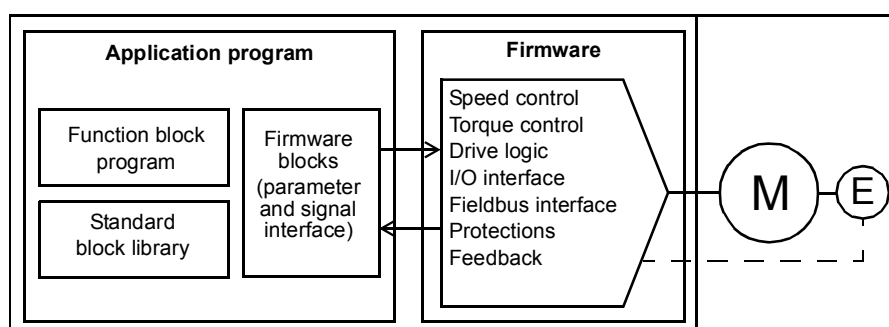
This chapter introduces drive programming and describes application programming using the DriveSPC PC tool.

General about drive programming

The drive control program is divided into two parts:

- firmware program
- application program.

Drive control program

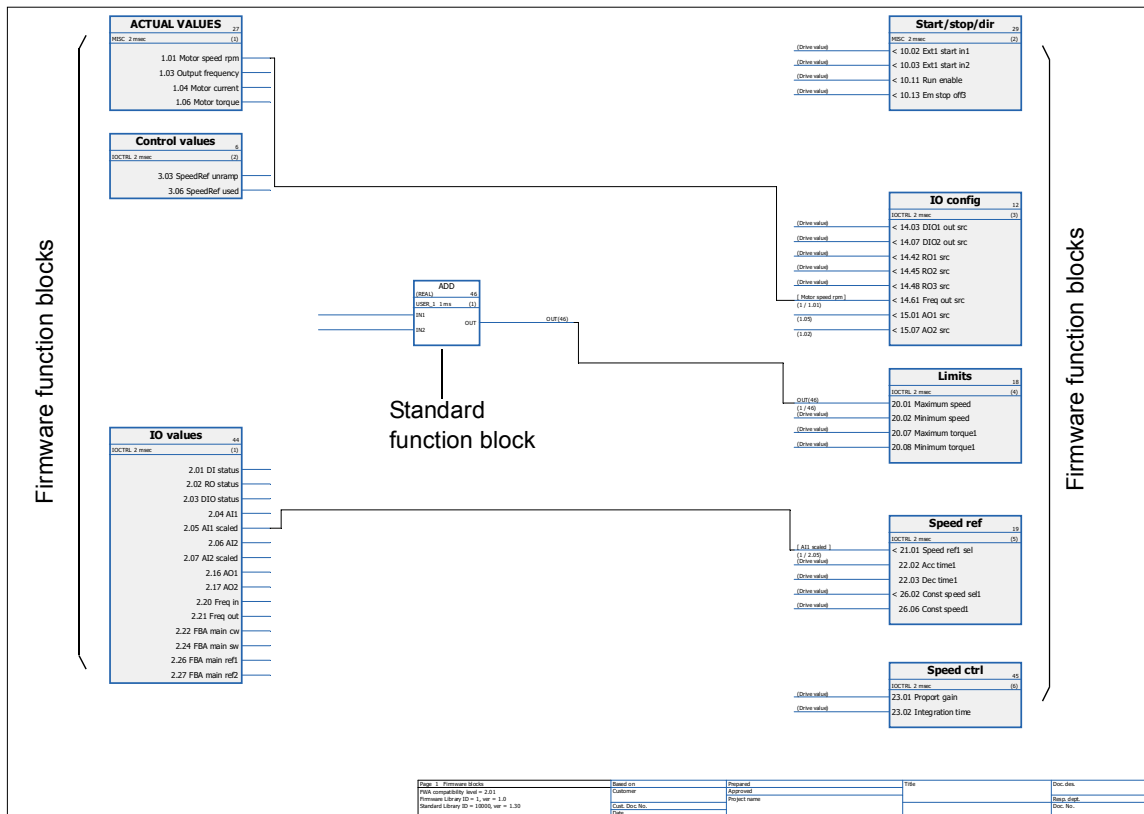


The firmware program performs the main control functions, including speed and torque control, drive logic (start/stop), I/O, feedback, communication and protection functions. Firmware functions are configured and programmed with parameters. Parameters can be set via the drive control panel, the DriveStudio PC tool or the fieldbus interface. For more information on programming via parameters, see the *Firmware manual*.

Application programming

The functions of the firmware program can be extended with application programming. The user can build an application program with firmware and standard functions blocks based on the IEC-61131 standard. ABB also offers customized application programs for specific applications; for more information, contact your local ABB representative.

Application programs are created with the DriveSPC PC tool. The following figure presents a view from DriveSPC.



Function blocks

An application program uses two types of function blocks: firmware function blocks and standard function blocks.

Firmware function blocks

The essential functions of the drive are represented as firmware function blocks in the DriveSPC PC tool. These blocks are part of the drive control firmware and act as an interface between the firmware and the application program. The inputs of the blocks correspond to drive parameters in groups 10...99 and can be modified via the application program; the outputs provide measured or calculated signals from groups 01...09. Note that not all parameters are accessible through the firmware function blocks.

The firmware function blocks available are presented in chapter [Firmware function blocks](#).

Standard function blocks

Standard function blocks (for example, ADD, AND) are used to create an executable application program. The maximum size of an application program is approximately 30 standard function blocks, depending on the block types used. The standard function blocks available are presented in chapter [Standard function blocks](#).

A standard function block library is always included in the drive delivery.

■ User parameters

User parameters can be created with the DriveSPC PC tool. User parameters can be added to any existing parameter group; the first available index is 70. Parameter groups 5 and 75...89 are available for user parameters starting from index 1. Using attributes, the parameters can be defined as write-protected, hidden, etc.

For more information, see *DriveSPC user manual* (3AFE68836590 [English]).

■ Application events

Application programmers can create their own application events (alarms and faults) by adding alarm and fault blocks; these blocks are managed through the Alarm and Fault Managers of the DriveSPC PC tool.

The operation of alarm and fault blocks is the same: when the block is enabled (by setting the Enable input to 1), an alarm or fault is generated by the drive.

■ Program execution

The application program is loaded to the permanent (non-volatile) memory of the memory unit (JMU). When the loading finishes, the drive control board is automatically reset and the downloaded program started. The program is executed in real time on the same Central Processing Unit (CPU of the drive control board) as the drive firmware. The program can be executed at the two dedicated time levels of 1 and 10 milliseconds, as well as other time levels between certain firmware tasks.

Note: Because the firmware and application programs use the same CPU, the programmer must ensure that the drive CPU is not overloaded. See parameter *01.21 Cpu usage*.

■ Application program licensing and protection

Note: This functionality is only available with DriveSPC version 1.5 and later.

The drive can be assigned an application licence consisting of an ID and password using the DriveSPC tool. Likewise, the application program created in DriveSPC can be protected by an ID and password. For instructions, see *DriveSPC user manual*.

If a protected application program is downloaded to a licensed drive, the IDs and passwords of the application and drive must match. A protected application cannot be downloaded to an unlicensed drive. On the other hand, an unprotected application can be downloaded to a licensed drive.

The ID of the application licence is displayed by DriveStudio in the drive software properties as APPL LICENCE. If the value is 0, no licence has been assigned to the drive.

Notes:

- The application licence can only be assigned to a complete drive, not a stand-alone control unit.
- The protected application can only be downloaded to a complete drive, not a stand-alone control unit.

■ **Operation modes**

The DriveSPC PC tool offers the following operation modes:

Off-line

When the off-line mode is used without a drive connection, the user can

- open an application program file (if it exists)
- modify and save the application program
- print the program pages.

When the off-line mode is used with a drive(s) connection, the user can

- connect the selected drive to DriveSPC
- upload an application program from the connected drive (an empty template which includes only the firmware blocks is available by default.)
- download the configured application program to the drive and start the program execution. The downloaded program contains the function block program and the parameter values set in DriveSPC.
- remove the program from the connected drive.

On-line

In the on-line mode, the user can

- modify firmware parameters (changes are stored directly to the drive memory)
 - modify application program parameters (that is, parameters created in DriveSPC)
 - monitor the actual values of all function blocks in real time.
-



Firmware function blocks

What this chapter contains

This chapter presents the firmware function blocks. The blocks are grouped according to parameter numbering in the drive firmware.

Description	Illustration
--------------------	---------------------

ACTUAL VALUES
(1)

<p>Basic signals for monitoring the drive.</p>	<table border="1" style="width: 100%;"> <tr> <td colspan="2" style="text-align: right;">ACTUAL VALUES</td> </tr> <tr> <td></td> <td style="text-align: right;">27</td> </tr> <tr> <td>MISC 2 msec</td> <td style="text-align: right;">(1)</td> </tr> <tr> <td colspan="2"> <ul style="list-style-type: none"> 1.01 Motor speed rpm 1.03 Output frequency 1.04 Motor current 1.06 Motor torque </td> </tr> </table>	ACTUAL VALUES			27	MISC 2 msec	(1)	<ul style="list-style-type: none"> 1.01 Motor speed rpm 1.03 Output frequency 1.04 Motor current 1.06 Motor torque 	
ACTUAL VALUES									
	27								
MISC 2 msec	(1)								
<ul style="list-style-type: none"> 1.01 Motor speed rpm 1.03 Output frequency 1.04 Motor current 1.06 Motor torque 									

IO values
(2)

<p>Input and output signals.</p>	<table border="1" style="width: 100%;"> <tr> <td colspan="2" style="text-align: right;">IO values</td> </tr> <tr> <td></td> <td style="text-align: right;">44</td> </tr> <tr> <td>IOCTRL 2 msec</td> <td style="text-align: right;">(1)</td> </tr> <tr> <td colspan="2"> <ul style="list-style-type: none"> 2.01 DI status 2.02 RO status 2.03 DIO status 2.04 AI1 2.05 AI1 scaled 2.06 AI2 2.07 AI2 scaled 2.16 AO1 2.17 AO2 2.20 Freq in 2.21 Freq out 2.22 FBA main cw 2.24 FBA main sw 2.26 FBA main ref1 2.27 FBA main ref2 </td> </tr> </table>	IO values			44	IOCTRL 2 msec	(1)	<ul style="list-style-type: none"> 2.01 DI status 2.02 RO status 2.03 DIO status 2.04 AI1 2.05 AI1 scaled 2.06 AI2 2.07 AI2 scaled 2.16 AO1 2.17 AO2 2.20 Freq in 2.21 Freq out 2.22 FBA main cw 2.24 FBA main sw 2.26 FBA main ref1 2.27 FBA main ref2 	
IO values									
	44								
IOCTRL 2 msec	(1)								
<ul style="list-style-type: none"> 2.01 DI status 2.02 RO status 2.03 DIO status 2.04 AI1 2.05 AI1 scaled 2.06 AI2 2.07 AI2 scaled 2.16 AO1 2.17 AO2 2.20 Freq in 2.21 Freq out 2.22 FBA main cw 2.24 FBA main sw 2.26 FBA main ref1 2.27 FBA main ref2 									

Control values
(3)

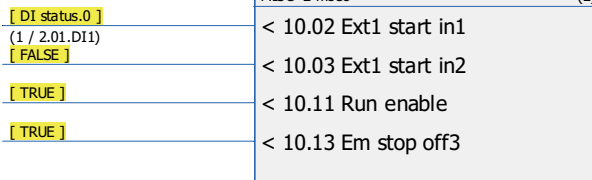
<p>Speed reference values.</p>	<table border="1" style="width: 100%;"> <tr> <td colspan="2" style="text-align: right;">Control values</td> </tr> <tr> <td></td> <td style="text-align: right;">6</td> </tr> <tr> <td>IOCTRL 2 msec</td> <td style="text-align: right;">(2)</td> </tr> <tr> <td colspan="2"> <ul style="list-style-type: none"> 3.03 SpeedRef unramp 3.06 SpeedRef used </td> </tr> </table>	Control values			6	IOCTRL 2 msec	(2)	<ul style="list-style-type: none"> 3.03 SpeedRef unramp 3.06 SpeedRef used 	
Control values									
	6								
IOCTRL 2 msec	(2)								
<ul style="list-style-type: none"> 3.03 SpeedRef unramp 3.06 SpeedRef used 									

Description	Illustration
-------------	--------------

Start/stop/dir

(10)

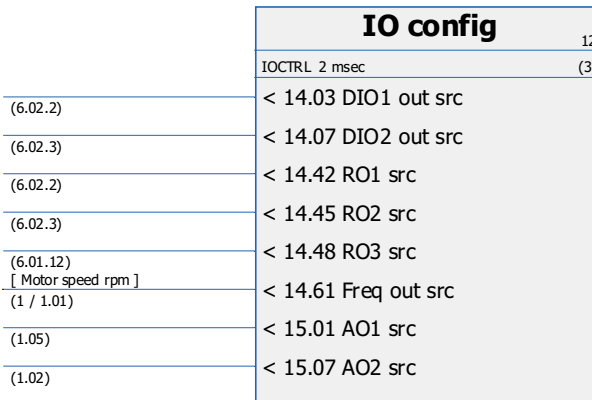
Source selections for start/stop, run enable and emergency stop signals.



IO config

(13)

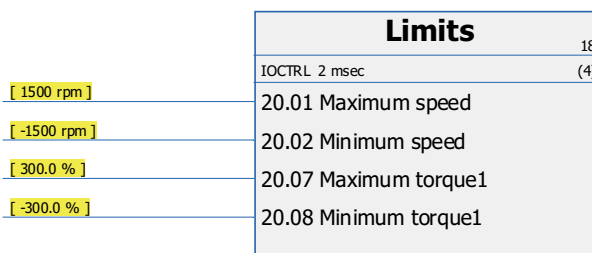
Configuration of digital input/outputs, relay outputs and analog outputs.



Limits

(20)

Drive operation limits.



Description	Illustration
-------------	--------------

Speed ref (21)

<p>Speed reference source selection; acceleration/ deceleration and constant speed settings.</p>	<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: auto;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="text-align: center;">Speed ref</th> <th style="text-align: right;">19</th> </tr> </thead> <tbody> <tr> <td colspan="2">IOCTRL 2 msec</td> <td style="text-align: right;">(5)</td> </tr> <tr> <td style="text-align: right;">[AI1 scaled] (1 / 2.05)</td> <td style="text-align: left;">< 21.01 Speed ref1 sel</td> <td></td> </tr> <tr> <td style="text-align: right;">[20.000 s]</td> <td style="text-align: left;">22.02 Acc time1</td> <td></td> </tr> <tr> <td style="text-align: right;">[20.000 s]</td> <td style="text-align: left;">22.03 Dec time1</td> <td></td> </tr> <tr> <td style="text-align: right;">[FALSE]</td> <td style="text-align: left;">< 26.02 Const speed sel1</td> <td></td> </tr> <tr> <td style="text-align: right;">[0 rpm]</td> <td style="text-align: left;">26.06 Const speed1</td> <td></td> </tr> </tbody> </table> </div>	Speed ref		19	IOCTRL 2 msec		(5)	[AI1 scaled] (1 / 2.05)	< 21.01 Speed ref1 sel		[20.000 s]	22.02 Acc time1		[20.000 s]	22.03 Dec time1		[FALSE]	< 26.02 Const speed sel1		[0 rpm]	26.06 Const speed1	
Speed ref		19																				
IOCTRL 2 msec		(5)																				
[AI1 scaled] (1 / 2.05)	< 21.01 Speed ref1 sel																					
[20.000 s]	22.02 Acc time1																					
[20.000 s]	22.03 Dec time1																					
[FALSE]	< 26.02 Const speed sel1																					
[0 rpm]	26.06 Const speed1																					

Speed ctrl (23)

<p>Speed controller settings.</p>	<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: auto;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="text-align: center;">Speed ctrl</th> <th style="text-align: right;">45</th> </tr> </thead> <tbody> <tr> <td colspan="2">IOCTRL 2 msec</td> <td style="text-align: right;">(6)</td> </tr> <tr> <td style="text-align: right;">[10.00]</td> <td style="text-align: left;">23.01 Proport gain</td> <td></td> </tr> <tr> <td style="text-align: right;">[0.500 s]</td> <td style="text-align: left;">23.02 Integration time</td> <td></td> </tr> </tbody> </table> </div>	Speed ctrl		45	IOCTRL 2 msec		(6)	[10.00]	23.01 Proport gain		[0.500 s]	23.02 Integration time	
Speed ctrl		45											
IOCTRL 2 msec		(6)											
[10.00]	23.01 Proport gain												
[0.500 s]	23.02 Integration time												

4

Standard function blocks

What this chapter contains

This chapter presents the standard function blocks. The blocks are grouped according to the grouping in the DriveSPC PC tool.

The number in brackets in the standard block heading is the block number.

Note: The given execution times can vary depending on the drive application used.

Terms

Data type	Description	Range
Boolean	Boolean	0 or 1
DINT	32-bit integer value (31 bits + sign)	-2147483648...2147483647
INT	16-bit integer value (15 bits + sign)	-32768...32767
PB	Packed Boolean	0 or 1 for each individual bit
REAL	$\underbrace{16\text{-bit value}}_{= \text{integer value}}$ $\underbrace{16\text{-bit value}}_{= \text{fractional value}}$ (31 bits + sign)	-32768,99998...32767,9998
REAL24	$\underbrace{8\text{-bit value}}_{= \text{integer value}}$ $\underbrace{24\text{-bit value}}_{= \text{fractional value}}$ (31 bits + sign)	-128,0...127,999

Alphabetical index

ABS	19	EXPT	20	NOT	23
ADD	19	FILT1	74	OR	23
AND	23	FILT2	74	PARRD	77
BGET	27	FIO_01_slot1	55	PARRDINTR	78
BITAND	27	FIO_01_slot2	55	PARRDPTR	78
BITOR	27	FIO_11_AI_slot1	56	PARWR	79
BOOL_TO_DINT	38	FIO_11_AI_slot2	58	PID	69
BOOL_TO_INT	39	FIO_11_AO_slot1	59	RAMP	70
BOP	80	FIO_11_AO_slot2	61	REAL24_TO_REAL	44
BSET	28	FIO_11_DIO_slot1	62	REALn_TO_DINT	44
CRITSPEED	64	FIO_11_DIO_slot2	63	REALn_TO_DINT_SIMP ..	44
CTD	46	FTRIG	52	REAL_TO_REAL24	43
CTD_DINT	46	FUNG-1V	66	REG	28
CTU	47	GE >=	36	REG-G	71
CTUD	49	GetBitPtr	77	ROL	24
CTUD_DINT	50	GetValPtr	77	ROR	24
CTU_DINT	48	GT >	36	RS	52
CYCLET	65	IF	81	RTRIG	53
D2D_Conf	31	INT	67	SEL	84
D2D_McastToken	31	INT_TO_BOOL	42	SHL	25
D2D_SendMessage	32	INT_TO_DINT	43	SHR	25
DATA_CONTAINER	65	LE <=	37	SOLUTION_FAULT	73
DEMUX-I	85	LEAD/LAG	76	SQRT	22
DEMUX-MI	85	LIMIT	83	SR	53
DINT_TO_BOOL	40	LT <	37	SR-D	29
DINT_TO_INT	40	MAX	83	SUB	22
DINT_TO_REALn	41	MIN	83	SWITCH	86
DINT_TO_REALn_SIMP ..	41	MOD	20	SWITCHC	86
DIV	19	MONO	87	TOF	88
DS_Read_Local	34	MOTPOT	68	TON	88
DS_WriteLocal	35	MOVE	21	TP	89
ELSE	80	MUL	21	XOR	26
ELSEIF	80	MULDIV	21		
ENDIF	81	MUX	84		
EQ	36	NE <>	37		

Arithmetic

ABS	
(10001)	
Illustration	
Execution time	0.53 μ s
Operation	The output (OUT) is the absolute value of the input (IN). $OUT = IN $
Inputs	The input data type is selected by the user. Input (IN): DINT, INT, REAL or REAL24
Outputs	Output (OUT): DINT, INT, REAL or REAL24

ADD	
(10000)	
Illustration	
Execution time	3.36 μ s (when two inputs are used) + 0.52 μ s (for every additional input). When all inputs are used, the execution time is 18.87 μ s.
Operation	The output (OUT) is the sum of the inputs (IN1...IN32). $OUT = IN1 + IN2 + \dots + IN32$ The output value is limited to the maximum and minimum values defined by the selected data type range.
Inputs	The input data type and the number of the inputs (2...32) are selected by the user. Input (IN1...IN32): DINT, INT, REAL or REAL24
Outputs	Output (OUT): DINT, INT, REAL or REAL24

DIV	
(10002)	
Illustration	
Execution time	2.55 μ s
Operation	The output (OUT) is input IN1 divided by input IN2. $OUT = IN1/IN2$ The output value is limited to the maximum and minimum values defined by the selected data type range. If the divider (IN2) is 0, the output is 0.

Inputs	The input data type is selected by the user. Input (IN1, IN2): INT, DINT, REAL, REAL24
Outputs	Output (OUT): INT, DINT, REAL, REAL24

EXPT (10003)	
Illustration	
Execution time	81.90 μ s
Operation	The output (OUT) is input IN1 raised to the power of the input IN2: $OUT = IN1^{IN2}$ If input IN1 is 0, the output is 0. The output value is limited to the maximum value defined by the selected data type range. Note: The execution of the EXPT function is slow.
Inputs	The input data type is selected by the user. Input (IN1): REAL, REAL24 Input (IN2): REAL
Outputs	Output (OUT): REAL, REAL24

MOD (10004)	
Illustration	
Execution time	1.67 μ s
Operation	The output (OUT) is the remainder of the division of the inputs IN1 and IN2. $OUT = \text{remainder of } IN1/IN2$ If input IN2 is zero, the output is zero.
Inputs	The input data type is selected by the user. Input (IN1, IN2): INT, DINT
Outputs	Output (OUT): INT, DINT

MOVE	
(10005)	
Illustration	
Execution time	2.10 μs (when two inputs are used) + 0.42 μs (for every additional input). When all inputs are used, the execution time is 14.55 μs.
Operation	Copies the input values (IN1...32) to the corresponding outputs (OUT1...32).
Inputs	The input data type and number of inputs (2...32) are selected by the user. Input (IN1...IN32): INT, DINT, REAL, REAL24, Boolean
Outputs	Output (OUT1...OUT32): INT, DINT, REAL, REAL24, Boolean

MUL	
(10006)	
Illustration	
Execution time	3.47 μs (when two inputs are used) + 2.28 μs (for every additional input). When all inputs are used, the execution time is 71.73 μs.
Operation	The output (OUT) is the product of the inputs (IN). $O = IN1 \times IN2 \times \dots \times IN32$ The output value is limited to the maximum and minimum values defined by the selected data type range.
Inputs	The input data type and the number of inputs (2...32) are selected by the user. Input (IN1...IN32): INT, DINT, REAL, REAL24
Outputs	Output (OUT): INT, DINT, REAL, REAL24

MULDIV	
(10007)	
Illustration	
Execution time	7.10 μs

22 Standard function blocks

Operation	<p>The output (O) is the product of input IN and input MUL divided by input DIV. $Output = (I \times MUL) / DIV$ O = whole value. REM = remainder value. Example: I = 2, MUL = 16 and DIV = 10: $(2 \times 16) / 10 = 3.2$, i.e. O = 3 and REM = 2 The output value is limited to the maximum and minimum values defined by the data type range.</p>
Inputs	<p>Input (I): DINT Multiplier input (MUL): DINT Divider input (DIV): DINT</p>
Outputs	<p>Output (O): DINT Remainder output (REM): DINT</p>

SQRT	
(10008)	
Illustration	
Execution time	2.09 μ s
Operation	<p>Output (OUT) is the square root of the input (IN). $OUT = \text{sqrt}(IN)$ Output is 0 if the input value is negative</p>
Inputs	<p>The input data type is selected by the user. Input (IN): REAL, REAL24</p>
Outputs	Output (OUT): REAL, REAL24

SUB	
(10009)	
Illustration	
Execution time	2.33 μ s
Operation	<p>Output (OUT) is the difference between the input signals (IN): $OUT = IN1 - IN2$ The output value is limited to the maximum and minimum values defined by the selected data type range.</p>
Inputs	<p>The input data type is selected by the user. Input (IN1, IN2): INT, DINT, REAL, REAL24</p>
Outputs	Output (OUT): INT, DINT, REAL, REAL24

Bitstring

AND (10010)																
Illustration																
Execution time	1.55 μ s (when two inputs are used) + 0.60 μ s (for every additional input). When all inputs are used, the execution time is 19.55 μ s.															
Operation	<p>The output (OUT) is 1 if all the connected inputs (IN1...IN32) are 1. Otherwise the output is 0.</p> <p>Truth table:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>IN1</th> <th>IN2</th> <th>OUT</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table> <p>The inputs can be inverted.</p>	IN1	IN2	OUT	0	0	0	0	1	0	1	0	0	1	1	1
IN1	IN2	OUT														
0	0	0														
0	1	0														
1	0	0														
1	1	1														
Inputs	The number of inputs is selected by the user. Input (IN1...IN32): Boolean															
Outputs	Output (OUT): Boolean															

NOT (10011)	
Illustration	
Execution time	0.32 μ s
Operation	The output (O) is 1 if the input (I) is 0. The output is 0 if the input is 1.
Inputs	Input (I): Boolean
Outputs	Output (O): Boolean

OR (10012)	
Illustration	
Execution time	1.55 μ s (when two inputs are used) + 0.60 μ s (for every additional input). When all inputs are used, the execution time is 19.55 μ s.

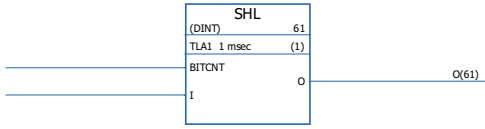
24 Standard function blocks

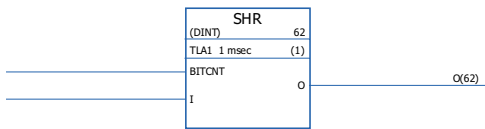
Operation	<p>The output (OUT) is 0, if all connected inputs (IN) are 0. Otherwise the output is 1.</p> <p>Truth table:</p> <table border="1"> <thead> <tr> <th>IN1</th> <th>IN2</th> <th>OUT</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table> <p>The inputs can be inverted.</p>	IN1	IN2	OUT	0	0	0	0	1	1	1	0	1	1	1	1
IN1	IN2	OUT														
0	0	0														
0	1	1														
1	0	1														
1	1	1														
Inputs	<p>The number of inputs (2...32) is selected by the user.</p> <p>Input (IN1...IN32): Boolean</p>															
Outputs	<p>Output (OUT): Boolean</p>															

ROL																																																																		
(10013)																																																																		
Illustration																																																																		
Execution time	1.28 μs																																																																	
Operation	<p>Input bits (I) are rotated to the left by the number (N) of bits defined by BITCNT. The N most significant bits (MSB) of the input are stored as the N least significant bits (LSB) of the output.</p> <p>Example: If BITCNT = 3</p> <table border="1"> <tr> <td>I</td> <td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td> </tr> <tr> <td>O</td> <td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td> </tr> </table> <p style="text-align: center;">3 MSB 3 LSB</p>	I	1	1	1	0	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0	1	0	1	O	0	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	1	0	1	1	1	1	1	1	1
I	1	1	1	0	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0	1	0	1																																		
O	0	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	1	0	1	1	1	1	1	1	1																																			
Inputs	<p>The input data type is selected by the user.</p> <p>Number of bits input (BITCNT): INT, DINT</p> <p>Input (I): INT, DINT</p>																																																																	
Outputs	<p>Output (O): INT, DINT</p>																																																																	

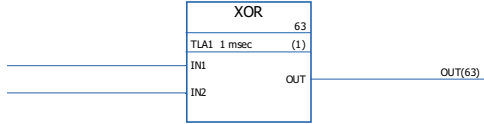
ROR	
(10014)	
Illustration	
Execution time	1.28 μs

<p>Operation</p>	<p>Input bits (I) are rotated to the right by the number (N) of bits defined by BITCNT. The N least significant bits (LSB) of the input are stored as the N most significant bits (MSB) of the output.</p> <p>Example: If BITCNT = 3</p> <table border="1" data-bbox="539 360 1430 562"> <tr> <td style="text-align: center;">I</td> <td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td> </tr> <tr> <td style="text-align: center;">O</td> <td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td> </tr> </table>	I	1	1	1	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0	1	0	1	O	1	0	1	1	1	1	0	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0
I	1	1	1	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0	1	0	1																																			
O	1	0	1	1	1	1	0	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0																																		
<p>Inputs</p>	<p>The input data type is selected by the user.</p> <p>Number of bits input (BITCNT): INT, DINT</p> <p>Input (I): INT, DINT</p>																																																																	
<p>Outputs</p>	<p>Output (O): INT, DINT</p>																																																																	

<p>SHL (10015)</p>																																																																		
<p>Illustration</p>																																																																		
<p>Execution time</p>	<p>0.80 µs</p>																																																																	
<p>Operation</p>	<p>Input bits (I) are rotated to the left by the number (N) of bits defined by BITCNT. The N most significant bits (MSB) of the input are lost and the N least significant bits (LSB) of the output are set to 0.</p> <p>Example: If BITCNT = 3</p> <table border="1" data-bbox="539 1272 1430 1458"> <tr> <td style="text-align: center;">I</td> <td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td> </tr> <tr> <td style="text-align: center;">O</td> <td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td> </tr> </table>	I	1	1	1	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0	1	0	1	O	0	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0	1	0	1	0	0	0
I	1	1	1	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0	1	0	1																																			
O	0	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0	1	0	1	0	0	0																																		
<p>Inputs</p>	<p>The input data type is selected by the user.</p> <p>Number of bits (BITCNT): INT; DINT</p> <p>Input (I): INT, DINT</p>																																																																	
<p>Outputs</p>	<p>Output (O): INT; DINT</p>																																																																	

<p>SHR (10016)</p>	
<p>Illustration</p>	
<p>Execution time</p>	<p>0.80 µs</p>

<p>Operation</p>	<p>Input bits (I) are rotated to the right by the number (N) of bits defined by BITCNT. The N least significant bits (LSB) of the input are lost and the N most significant bits (MSB) of the output are set to 0.</p> <p>Example: If BITCNT = 3</p> <table border="1" data-bbox="391 358 1276 560"> <tr> <td style="text-align: right;">I</td> <td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td> </tr> <tr> <td style="text-align: right;">O</td> <td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td> </tr> </table> <p style="text-align: right;">3 LSB</p> <p style="text-align: left;">3 MSB</p>	I	1	1	1	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0	1	0	1	O	0	0	0	1	1	1	0	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0
I	1	1	1	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0	1	0	1																																			
O	0	0	0	1	1	1	0	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	0	0	1	1	0	0	1	1	0																																		
<p>Inputs</p>	<p>The input data type is selected by the user.</p> <p>Number of bits (BITCNT): INT; DINT</p> <p>Input (I): INT, DINT</p>																																																																	
<p>Outputs</p>	<p>Output (O): INT; DINT</p>																																																																	

<p>XOR</p> <p>(10017)</p>																
<p>Illustration</p>																
<p>Execution time</p>	<p>1.24 μs (when two inputs are used) + 0.72 μs (for every additional input). When all inputs are used, the execution time is 22.85 μs.</p>															
<p>Operation</p>	<p>The output (OUT) is 1 if one of the connected inputs (IN1...IN32) is 1. Output is zero if all the inputs have the same value.</p> <p>Example:</p> <table border="1" data-bbox="383 1276 861 1478"> <thead> <tr> <th>IN1</th> <th>IN2</th> <th>OUT</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table> <p>The inputs can be inverted.</p>	IN1	IN2	OUT	0	0	0	0	1	1	1	0	1	1	1	0
IN1	IN2	OUT														
0	0	0														
0	1	1														
1	0	1														
1	1	0														
<p>Inputs</p>	<p>The number of inputs (2...32) is selected by the user.</p> <p>Input (IN1...IN32): Boolean</p>															
<p>Outputs</p>	<p>Output (OUT): Boolean</p>															

Bitwise

BGET (10034)	
Illustration	
Execution time	0.88 μ s
Operation	The output (O) is the value of the selected bit (BITNR) of the input (I). BITNR: Bit number (0 = bit number 0, 31 = bit number 31) If bit number is not in the range of 0...31 (for DINT) or 0...15 (for INT), the output is 0.
Inputs	The input data type is selected by the user. Number of the bit (BITNR): DINT Input (I): DINT, INT
Outputs	Output (O): Boolean

BITAND (10035)							
Illustration							
Execution time	0.32 μ s						
Operation	The output (O) bit value is 1 if the corresponding bit values of the inputs (I1 and I2) are 1. Otherwise the output bit value is 0. Example:						
	<table border="1" style="width: 100%; text-align: center;"> <tbody> <tr> <td style="width: 5%;">I1</td> <td>1 1 1 0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1</td> </tr> <tr> <td>I2</td> <td>0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 1</td> </tr> <tr> <td>O</td> <td>0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0 1 0 0 1 0 0 1 0 1</td> </tr> </tbody> </table>	I1	1 1 1 0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1	I2	0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 1	O	0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0 1 0 0 1 0 0 1 0 1
I1	1 1 1 0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1						
I2	0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 1						
O	0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0 1 0 0 1 0 0 1 0 1						
Inputs	Input (I1, I2): DINT						
Outputs	Output (O): DINT						

BITOR (10036)	
Illustration	
Execution time	0.32 μ s

Operation	<p>The output (O) bit value is 1 if the corresponding bit value of any of the inputs (I1 or I2) is 1. Otherwise the output bit value is 0.</p> <p>Example:</p> <table border="1"> <tr> <td>I1</td> <td>1 1 1 0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1</td> </tr> <tr> <td>I2</td> <td>0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 1</td> </tr> <tr> <td>O</td> <td>1 1 1 0 0 1 1 1 1 1 1 0 1 1 1 1 1 1 0 1 1 0 1 1 1 0 1 1 1 1 1 1</td> </tr> </table>	I1	1 1 1 0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1	I2	0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 1	O	1 1 1 0 0 1 1 1 1 1 1 0 1 1 1 1 1 1 0 1 1 0 1 1 1 0 1 1 1 1 1 1
I1	1 1 1 0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1						
I2	0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 1						
O	1 1 1 0 0 1 1 1 1 1 1 0 1 1 1 1 1 1 0 1 1 0 1 1 1 0 1 1 1 1 1 1						
Inputs	Input (I1, I2): DINT						
Outputs	Output (O): DINT						

BSET (10037)	
Illustration	<p>The diagram shows a rectangular block labeled 'BSET'. At the top left, it specifies '(DINT) 67'. Below that, 'TLA1 1 msec (1)'. On the left side, there are four input lines labeled 'EN', 'BITNR', 'BIT', and 'I'. On the right side, there is one output line labeled 'O(67)'.</p>
Execution time	1.36 μs
Operation	<p>The value of a selected bit (BITNR) of the input (I) is set as defined by the bit value input (BIT). The function must be enabled by the enable input (EN).</p> <p>BITNR: Bit number (0 = bit number 0, 31 = bit number 31)</p> <p>If BITNR is not in the range of 0...31 (for DINT) or 0...15 (for INT) or if EN is reset to zero, the input value is stored to the output as it is (i.e. no bit setting occurs).</p> <p>Example: EN = 1, BITNR = 3, BIT = 0 IN = 0000 0000 1111 1111 O = 0000 0000 1111 0111</p>
Inputs	<p>The input data type is selected by the user.</p> <p>Enable input (EN): Boolean</p> <p>Number of the bit (BITNR): DINT</p> <p>Bit value input (BIT): Boolean</p> <p>Input (I): INT, DINT</p>
Outputs	Output (O): INT, DINT

REG (10038)	
Illustration	<p>The diagram shows a rectangular block labeled 'REG'. At the top left, it specifies '(BOOL) 68'. Below that, 'TLA1 1 msec (1)'. On the left side, there are five input lines labeled 'S', '>L', 'R', 'I1', and 'I2'. On the right side, there are two output lines labeled 'O1(68)' and 'O2(68)'.</p>

Execution time	2.27 μ s (when two inputs are used) + 1.02 μ s (for every additional input). When all inputs are used, the execution time is 32.87 μ s.																																																						
Operation	<p>The input (I1...I32) value is stored to the corresponding output (O1...O32) if the load input (L) is set to 1 or the set input (S) is 1. When the load input is set to 1, the input value is stored to the output only once. When the set input is 1, the input value is stored to the output every time the block is executed. The set input overrides the load input.</p> <p>If the reset input (R) is 1, all connected outputs are 0.</p> <p>Example:</p> <table border="1"> <thead> <tr> <th>S</th> <th>R</th> <th>L</th> <th>I</th> <th>O1_{previous}</th> <th>O1</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>10</td> <td>15</td> <td>15</td> </tr> <tr> <td>0</td> <td>0</td> <td>0->1</td> <td>20</td> <td>15</td> <td>20</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>30</td> <td>20</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0->1</td> <td>40</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>50</td> <td>0</td> <td>50</td> </tr> <tr> <td>1</td> <td>0</td> <td>0->1</td> <td>60</td> <td>50</td> <td>60</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>70</td> <td>60</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0->1</td> <td>80</td> <td>0</td> <td>0</td> </tr> </tbody> </table> <p>O1_{previous} is the previous cycle output value.</p>	S	R	L	I	O1 _{previous}	O1	0	0	0	10	15	15	0	0	0->1	20	15	20	0	1	0	30	20	0	0	1	0->1	40	0	0	1	0	0	50	0	50	1	0	0->1	60	50	60	1	1	0	70	60	0	1	1	0->1	80	0	0
S	R	L	I	O1 _{previous}	O1																																																		
0	0	0	10	15	15																																																		
0	0	0->1	20	15	20																																																		
0	1	0	30	20	0																																																		
0	1	0->1	40	0	0																																																		
1	0	0	50	0	50																																																		
1	0	0->1	60	50	60																																																		
1	1	0	70	60	0																																																		
1	1	0->1	80	0	0																																																		
Inputs	<p>The input data type and number of inputs (1...32) are selected by the user.</p> <p>Set input (S): Boolean</p> <p>Load input (L): Boolean</p> <p>Reset input (R): Boolean</p> <p>Input (I1...I32): Boolean, INT, DINT, REAL, REAL24</p>																																																						
Outputs	Output (O1...O32): Boolean, INT, DINT, REAL, REAL24																																																						

SR-D	
(10039)	
Illustration	
Execution time	1.04 μ s

<p>Operation</p>	<p>When clock input (C) is set to 1, the data input (D) value is stored to the output (O). When reset input (R) is set to 1, the output is set to 0.</p> <p>If only set (S) and reset (R) inputs are used, SR-D block acts as an SR block: The output is 1 if the set input (S) is 1. The output will retain the previous output state if the set input (S) and reset input (R) are 0. The output is 0 if the set input is 0 and the reset input is 1.</p> <p>Truth table:</p> <table border="1" data-bbox="383 465 1289 1227"> <thead> <tr> <th>S</th> <th>R</th> <th>D</th> <th>C</th> <th>O_{previous}</th> <th>O</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0 (= Previous output value)</td> </tr> <tr> <td>0</td> <td>0</td> <td>0</td> <td>0 -> 1</td> <td>0</td> <td>0 (= Data input value)</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>0</td> <td>0</td> <td>0 (= Previous output value)</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>0 -> 1</td> <td>0</td> <td>1 (= Data input value)</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>0</td> <td>1</td> <td>0 (Reset)</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>0 -> 1</td> <td>0</td> <td>0 (Reset)</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> <td>0 (Reset)</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0 -> 1</td> <td>0</td> <td>0 (Reset)</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>1 (= Set value)</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>0 -> 1</td> <td>1</td> <td>0 (= Data input value) for one execution cycle, then changes to 1 according to the set input (S = 1).</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0</td> <td>1</td> <td>1 (= Set value)</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0 -> 1</td> <td>1</td> <td>1 (= Data input value)</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>0</td> <td>1</td> <td>0 (Reset)</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>0 -> 1</td> <td>0</td> <td>0 (Reset)</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> <td>0 (Reset)</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>0 -> 1</td> <td>0</td> <td>0 (Reset)</td> </tr> </tbody> </table> <p>O_{previous} is the previous cycle output value.</p>	S	R	D	C	O _{previous}	O	0	0	0	0	0	0 (= Previous output value)	0	0	0	0 -> 1	0	0 (= Data input value)	0	0	1	0	0	0 (= Previous output value)	0	0	1	0 -> 1	0	1 (= Data input value)	0	1	0	0	1	0 (Reset)	0	1	0	0 -> 1	0	0 (Reset)	0	1	1	0	0	0 (Reset)	0	1	1	0 -> 1	0	0 (Reset)	1	0	0	0	0	1 (= Set value)	1	0	0	0 -> 1	1	0 (= Data input value) for one execution cycle, then changes to 1 according to the set input (S = 1).	1	0	1	0	1	1 (= Set value)	1	0	1	0 -> 1	1	1 (= Data input value)	1	1	0	0	1	0 (Reset)	1	1	0	0 -> 1	0	0 (Reset)	1	1	1	0	0	0 (Reset)	1	1	1	0 -> 1	0	0 (Reset)
S	R	D	C	O _{previous}	O																																																																																																		
0	0	0	0	0	0 (= Previous output value)																																																																																																		
0	0	0	0 -> 1	0	0 (= Data input value)																																																																																																		
0	0	1	0	0	0 (= Previous output value)																																																																																																		
0	0	1	0 -> 1	0	1 (= Data input value)																																																																																																		
0	1	0	0	1	0 (Reset)																																																																																																		
0	1	0	0 -> 1	0	0 (Reset)																																																																																																		
0	1	1	0	0	0 (Reset)																																																																																																		
0	1	1	0 -> 1	0	0 (Reset)																																																																																																		
1	0	0	0	0	1 (= Set value)																																																																																																		
1	0	0	0 -> 1	1	0 (= Data input value) for one execution cycle, then changes to 1 according to the set input (S = 1).																																																																																																		
1	0	1	0	1	1 (= Set value)																																																																																																		
1	0	1	0 -> 1	1	1 (= Data input value)																																																																																																		
1	1	0	0	1	0 (Reset)																																																																																																		
1	1	0	0 -> 1	0	0 (Reset)																																																																																																		
1	1	1	0	0	0 (Reset)																																																																																																		
1	1	1	0 -> 1	0	0 (Reset)																																																																																																		
<p>Inputs</p>	<p>Set input (S): Boolean Data input (D): Boolean Clock input (C): Boolean Reset input (R): Boolean</p>																																																																																																						
<p>Outputs</p>	<p>Output (O): Boolean</p>																																																																																																						

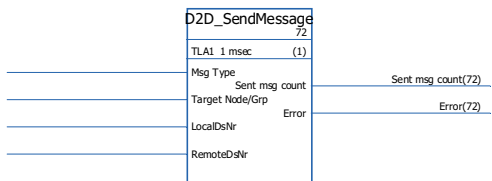
Communication

See also section [Drive-to-drive communication](#) on page 95.

D2D_Conf																			
(10092)																			
Illustration	<p>The diagram shows a block named 'D2D_Conf' with a reference number '70'. It has three input lines on the left: 'Ref1 Cycle Sel', 'Ref2 Cycle Sel', and 'Std Mcast Group'. It has one output line on the right labeled 'Error' with a reference number '(70)'. A top-left corner label indicates 'TLA1 1 msec (1)'.</p>																		
Execution time	-																		
Operation	<p>Defines handling interval for drive-to-drive references 1 and 2, and the address (group number) for standard (non-chained) multicast messages.</p> <p>The values of the Ref1/2 Cycle Sel inputs correspond to the following intervals:</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Handling interval</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Default (500 µs for reference 1; 2 ms for reference 2)</td> </tr> <tr> <td>1</td> <td>250 µs</td> </tr> <tr> <td>2</td> <td>500 µs</td> </tr> <tr> <td>3</td> <td>2 ms</td> </tr> </tbody> </table> <p>Note: Negative value of Ref2 Cycle Sel disables the handling of Ref2 (if disabled in the master, it must be disabled in all follower drives as well).</p> <p>Allowable values for the Std Mcast Group input are 0 (= multicasting not used) and 1...62 (multicast group).</p> <p>An unconnected input, or an input in an error state, is interpreted as having the value 0.</p> <p>The error codes indicated by the Error output are as follows:</p> <table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>REF1_CYCLE_ERR: Value of input Ref1 Cycle Sel out of range</td> </tr> <tr> <td>1</td> <td>REF2_CYCLE_ERR: Value of input Ref2 Cycle Sel out of range</td> </tr> <tr> <td>2</td> <td>STD_MCAST_ERR: Value of input Std Mcast Group out of range</td> </tr> </tbody> </table>	Value	Handling interval	0	Default (500 µs for reference 1; 2 ms for reference 2)	1	250 µs	2	500 µs	3	2 ms	Bit	Description	0	REF1_CYCLE_ERR: Value of input Ref1 Cycle Sel out of range	1	REF2_CYCLE_ERR: Value of input Ref2 Cycle Sel out of range	2	STD_MCAST_ERR: Value of input Std Mcast Group out of range
Value	Handling interval																		
0	Default (500 µs for reference 1; 2 ms for reference 2)																		
1	250 µs																		
2	500 µs																		
3	2 ms																		
Bit	Description																		
0	REF1_CYCLE_ERR: Value of input Ref1 Cycle Sel out of range																		
1	REF2_CYCLE_ERR: Value of input Ref2 Cycle Sel out of range																		
2	STD_MCAST_ERR: Value of input Std Mcast Group out of range																		
Inputs	<p>Drive-to-drive reference 1 handling interval (Ref1 Cycle Sel): INT</p> <p>Drive-to-drive reference 2 handling interval (Ref2 Cycle Sel): INT</p> <p>Standard multicast address (Std Mcast Group): INT</p>																		
Outputs	Error output (Error): PB																		

D2D_McastToken	
(10096)	
Illustration	<p>The diagram shows a block named 'D2D_McastToken' with a reference number '71'. It has two input lines on the left: 'Target Node' and 'Mcast Cycle'. It has one output line on the right labeled 'Error' with a reference number '(71)'. A top-left corner label indicates 'TLA1 1 msec (1)'.</p>
Execution time	-

<p>Operation</p>	<p>Configures the transmission of token messages sent to a follower. Each token authorizes the follower to send one message to another follower or group of followers. For the message types, see block D2D_SendMessage.</p> <p>Note: This block is only supported in the master.</p> <p>The Target Node input defines the node address the master sends the tokens to; the range is 1...62.</p> <p>The Mcast Cycle specifies the interval between token messages in the range of 2...1000 milliseconds. Setting this input to 0 disables the sending of tokens.</p> <p>The error codes indicated by the Error output are as follows:</p> <table border="1" data-bbox="383 537 1286 797"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>D2D_MODE_ERR: Drive is not master</td> </tr> <tr> <td>5</td> <td>TOO_SHORT_CYCLE: Token interval is too short, causing overloading</td> </tr> <tr> <td>6</td> <td>INVALID_INPUT_VAL: An input value is out of range</td> </tr> <tr> <td>7</td> <td>GENERAL_D2D_ERR: Drive-to-drive communication driver failed to initialize message</td> </tr> </tbody> </table>	Bit	Description	0	D2D_MODE_ERR: Drive is not master	5	TOO_SHORT_CYCLE: Token interval is too short, causing overloading	6	INVALID_INPUT_VAL: An input value is out of range	7	GENERAL_D2D_ERR: Drive-to-drive communication driver failed to initialize message
Bit	Description										
0	D2D_MODE_ERR: Drive is not master										
5	TOO_SHORT_CYCLE: Token interval is too short, causing overloading										
6	INVALID_INPUT_VAL: An input value is out of range										
7	GENERAL_D2D_ERR: Drive-to-drive communication driver failed to initialize message										
<p>Inputs</p>	<p>Token recipient (Target Node): INT</p> <p>Token interval (Mcast Cycle): INT</p>										
<p>Outputs</p>	<p>Error output (Error): DINT</p>										

<p>D2D_SendMessage (10095)</p>	
<p>Illustration</p>	 <p>The diagram shows a block titled 'D2D_SendMessage' with a '72' in a box at the top right. On the left side, there are four input lines: 'Msg Type', 'Target Node/Grp', 'LocalDsNr', and 'RemoteDsNr'. On the right side, there are two output lines: 'Sent msg count(72)' and 'Error(72)'. A note 'TLAI 1 msec (1)' is located near the top left of the block.</p>
<p>Execution time</p>	<p>-</p>

Operation	Configures the transmission between the dataset tables of drives. The Msg Type input defines the message type as follows:	
	Value	Message type
	0	Disabled
	1	<p>Master P2P:</p> <p>The master sends the contents of a local dataset (specified by LocalDsNr input) to the dataset table (dataset number specified by RemoteDsNr input) of a follower (specified by Target Node/Grp input).</p> <p>The follower replies by sending the next dataset (RemoteDsNr + 1) to the master (LocalDsNr + 1).</p> <p>The node number of a drive is defined by parameter 57.03.</p> <p>Note: Only supported in the master drive.</p>
	2	<p>Read Remote:</p> <p>The master reads a dataset (specified by RemoteDsNr input) from a follower (specified by Target Node/Grp input) and stores it into local dataset table (dataset number specified by LocalDsNr input).</p> <p>The node number of a drive is defined by parameter 57.03.</p> <p>Note: Only supported in the master drive.</p>
	3	<p>Follower P2P:</p> <p>The follower sends the contents of a local dataset (specified by LocalDsNr input) to the dataset table (dataset number specified by RemoteDsNr input) of another follower (specified by Target Node/Grp input).</p> <p>The node number of a drive is defined by parameter 57.03.</p> <p>Note: Only supported in a follower drive. A token from the master drive is required for the follower to be able to send the message. See block D2D_McastToken.</p>
	4	<p>Standard Multicast:</p> <p>The drive sends the contents of a local dataset (specified by LocalDsNr input) to the dataset table (dataset number specified by RemoteDsNr input) of a group of followers (specified by Target Node/Grp input).</p> <p>Which multicast group a drive belongs to is defined by the Std Mcast Group input of the D2D_Conf block.</p> <p>A token from the master drive is required for a follower to be able to send the message. See the block D2D_McastToken.</p>
	5	<p>Broadcast:</p> <p>The drive sends the contents of a local dataset (specified by LocalDsNr input) to the dataset table (dataset number specified by RemoteDsNr input) of all followers.</p> <p>A token from the master drive is required for a follower to be able to send the message. See block D2D_McastToken.</p> <p>Note: With this message type, the Target Node/Grp input must be connected in DriveSPC even if not used.</p>
<p>The Target Node/Grp input specifies the target drive or multicast group of drives depending on message type. See the message type explanations above.</p> <p>Note: The input must be connected in DriveSPC even if not used.</p> <p>The LocalDsNr input specifies the number of the local dataset used as the source or the target of the message.</p> <p>The RemoteDsNr input specifies the number of the remote dataset used as the target or the source of the message.</p> <p>The Sent msg count output is a wrap-around counter of successfully sent messages.</p>		

	<p>The error codes indicated by the Error output are as follows:</p> <table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>D2D_MODE_ERR: Drive-to-drive communication not activated, or message type not supported in current drive-to-drive mode (master/follower)</td> </tr> <tr> <td>1</td> <td>LOCAL_DS_ERR: LocalDsNr input out of range (16...199)</td> </tr> <tr> <td>2</td> <td>TARGET_NODE_ERR: Target Node/Grp input out of range (1...62)</td> </tr> <tr> <td>3</td> <td>REMOTE_DS_ERR: Remote dataset number out of range (16...199)</td> </tr> <tr> <td>4</td> <td>MSG_TYPE_ERR: Msg Type input out of range (0...5)</td> </tr> <tr> <td>5...6</td> <td>Reserved</td> </tr> <tr> <td>7</td> <td>GENERAL_D2D_ERR: Unspecified error in D2D driver</td> </tr> <tr> <td>8</td> <td>RESPONSE_ERR: Syntax error in received response</td> </tr> <tr> <td>9</td> <td>TRA_PENDING: Message has not yet been sent</td> </tr> <tr> <td>10</td> <td>REC_PENDING: Response has not yet been received</td> </tr> <tr> <td>11</td> <td>REC_TIMEOUT: No response received</td> </tr> <tr> <td>12</td> <td>REC_ERROR: Frame error in received message</td> </tr> <tr> <td>13</td> <td>REJECTED: Message has been removed from transmit buffer</td> </tr> <tr> <td>14</td> <td>BUFFER_FULL: Transmit buffer full</td> </tr> </tbody> </table>	Bit	Description	0	D2D_MODE_ERR: Drive-to-drive communication not activated, or message type not supported in current drive-to-drive mode (master/follower)	1	LOCAL_DS_ERR: LocalDsNr input out of range (16...199)	2	TARGET_NODE_ERR: Target Node/Grp input out of range (1...62)	3	REMOTE_DS_ERR: Remote dataset number out of range (16...199)	4	MSG_TYPE_ERR: Msg Type input out of range (0...5)	5...6	Reserved	7	GENERAL_D2D_ERR: Unspecified error in D2D driver	8	RESPONSE_ERR: Syntax error in received response	9	TRA_PENDING: Message has not yet been sent	10	REC_PENDING: Response has not yet been received	11	REC_TIMEOUT: No response received	12	REC_ERROR: Frame error in received message	13	REJECTED: Message has been removed from transmit buffer	14	BUFFER_FULL: Transmit buffer full
Bit	Description																														
0	D2D_MODE_ERR: Drive-to-drive communication not activated, or message type not supported in current drive-to-drive mode (master/follower)																														
1	LOCAL_DS_ERR: LocalDsNr input out of range (16...199)																														
2	TARGET_NODE_ERR: Target Node/Grp input out of range (1...62)																														
3	REMOTE_DS_ERR: Remote dataset number out of range (16...199)																														
4	MSG_TYPE_ERR: Msg Type input out of range (0...5)																														
5...6	Reserved																														
7	GENERAL_D2D_ERR: Unspecified error in D2D driver																														
8	RESPONSE_ERR: Syntax error in received response																														
9	TRA_PENDING: Message has not yet been sent																														
10	REC_PENDING: Response has not yet been received																														
11	REC_TIMEOUT: No response received																														
12	REC_ERROR: Frame error in received message																														
13	REJECTED: Message has been removed from transmit buffer																														
14	BUFFER_FULL: Transmit buffer full																														
Inputs	<p>Message type (Msg Type): INT Target node or multicast group (Target Node/Grp): INT Local dataset number (LocalDsNr): INT Remote dataset number (RemoteDsNr): INT</p>																														
Outputs	<p>Successfully sent messages counter (Sent msg count): DINT Error output (Error): PB</p>																														

DS_Read_Local					
(10094)					
Illustration					
Execution time	-				
Operation	<p>Reads the dataset defined by the LocalDsNr input from the local dataset table. One dataset contains one 16-bit and one 32-bit word which are directed to the Data1 16B and Data2 32B outputs respectively.</p> <p>The LocalDsNr input defines the number of the dataset to be read.</p> <p>The error codes indicated by the Error output are as follows:</p> <table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>LOCAL_DS_ERR: LocalDsNr out of range (16...199)</td> </tr> </tbody> </table>	Bit	Description	1	LOCAL_DS_ERR: LocalDsNr out of range (16...199)
Bit	Description				
1	LOCAL_DS_ERR: LocalDsNr out of range (16...199)				
Inputs	Local dataset number (LocalDsNr): INT				
Outputs	<p>Contents of dataset (Data1 16B): INT Contents of dataset (Data2 32B): DINT Error output (Error): DINT</p>				

DS_WriteLocal					
(10093)					
Illustration					
Execution time	-				
Operation	<p>Writes data into the local dataset table. Each dataset contains 48 bits; the data is input through the Data1 16B (16 bits) and Data2 32B (32 bits) inputs. The dataset number is defined by the LocalDsNr input.</p> <p>The error codes indicated by the Error output are as follows:</p> <table border="1" style="width: 100%;"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>LOCAL_DS_ERR: LocalDsNr out of range (16...199)</td> </tr> </tbody> </table>	Bit	Description	1	LOCAL_DS_ERR: LocalDsNr out of range (16...199)
Bit	Description				
1	LOCAL_DS_ERR: LocalDsNr out of range (16...199)				
Inputs	<p>Local dataset number (LocalDsNr): INT Contents of dataset (Data1 16B): INT Contents of dataset (Data2 32B): DINT</p>				
Outputs	Error output (Error): DINT				

Comparison

EQ	
(10040)	
Illustration	
Execution time	0.89 μ s (when two inputs are used) + 0.43 μ s (for every additional input). When all inputs are used, the execution time is 13.87 μ s.
Operation	The output (OUT) is 1 if all the connected input values are equal (IN1 = IN2 = ... = IN32). Otherwise the output is 0.
Inputs	The input data type and the number of inputs (2...32) are selected by the user. Input (IN1...IN32): INT, DINT, REAL, REAL24
Outputs	Output (OUT): Boolean

GE >=	
(10041)	
Illustration	
Execution time	0.89 μ s (when two inputs are used) + 0.43 μ s (for every additional input). When all inputs are used, the execution time is 13.87 μ s.
Operation	The output (OUT) is 1 if (IN1 > IN2) & (IN2 > IN3) & ... & (IN31 > IN32). Otherwise the output is 0.
Inputs	The input data type and the number of inputs (2...32) are selected by the user. Input (IN1...IN32): INT, DINT, REAL, REAL24
Outputs	Output (OUT): Boolean

GT >	
(10042)	
Illustration	
Execution time	0.89 μ s (when two inputs are used) + 0.43 μ s (for every additional input). When all inputs are used, the execution time is 13.87 μ s.
Operation	The output (OUT) is 1 if (IN1 > IN2) & (IN2 > IN3) & ... & (IN31 > IN32). Otherwise the output is 0.
Inputs	The input data type and the number of inputs (2...32) are selected by the user. Input (IN1...IN32): INT, DINT, REAL, REAL24

Outputs	Output (OUT): Boolean
----------------	-----------------------

LE <= (10043)	
Illustration	
Execution time	0.89 μ s (when two inputs are used) + 0.43 μ s (for every additional input). When all inputs are used, the execution time is 13.87 μ s.
Operation	Output (OUT) is 1 if $(IN1 \leq IN2) \& (IN2 \leq IN3) \& \dots \& (IN31 \leq IN32)$. Otherwise the output is 0.
Inputs	The input data type and the number of inputs (2...32) are selected by the user. Input (IN1...IN32): INT, DINT, REAL, REAL24
Outputs	Output (OUT): Boolean

LT < (10044)	
Illustration	
Execution time	0.89 μ s (when two inputs are used) + 0.43 μ s (for every additional input). When all inputs are used, the execution time is 13.87 μ s.
Operation	Output (OUT) is 1 if $(IN1 < IN2) \& (IN2 < IN3) \& \dots \& (IN31 < IN32)$. Otherwise the output is 0.
Inputs	The input data type and the number of inputs (2...32) are selected by the user. Input (IN1...IN32): INT, DINT, REAL, REAL24
Outputs	Output (OUT): Boolean

NE <> (10045)	
Illustration	
Execution time	0.44 μ s
Operation	The output (O) is 1 if $I1 \neq I2$. Otherwise the output is 0.
Inputs	The input data type is selected by the user. Input (I1, I2): INT, DINT, REAL, REAL24
Outputs	Output (O): Boolean

Conversion

BOOL_TO_DINT (10018)	
Illustration	<p>The diagram shows a function block named 'BOOL_TO_DINT' with a width of 81 units. It has 32 inputs on the left side, labeled IN1 through IN31, and one input labeled SIGN. The output is labeled OUT(81). A parameter 'TLAI 1 msec (1)' is located at the top right of the block.</p>
Execution time	13.47 μs
Operation	<p>The output (OUT) value is a 32-bit integer value formed from the boolean input (IN1...IN31 and SIGN) values. IN1 = bit 0 and IN31 = bit 30.</p> <p>Example:</p> <p>IN1 = 1, IN2 = 0, IN3...IN31 = 1, SIGN = 1</p> <p>OUT = 1111 1111 1111 1111 1111 1111 1101</p> <p style="text-align: center;"> SIGN IN31...IN1 </p>
Inputs	Sign input (SIGN): Boolean Input (IN1...IN31): Boolean
Outputs	Output (OUT): DINT (31 bits + sign)

BOOL_TO_INT (10019)	
Illustration	
Execution time	5.00 µs
Operation	The output (OUT) value is a 16-bit integer value formed from the boolean input (IN1...IN15 and SIGN) values. IN1 = bit 0 and IN15 = bit 14. Example: IN1...IN15 = 1, SIGN = 0 OUT = 0111 1111 1111 1111 SIGN IN15...IN1
Inputs	Input (IN1...IN15): Boolean Sign input (SIGN): Boolean
Outputs	Output (OUT): DINT (15 bits + sign)

DINT_TO_BOOL	
(10020)	
Illustration	
Execution time	11.98 µs
Operation	<p>The boolean output (OUT1...OUT32) values are formed from the 32-bit integer input (IN) value.</p> <p>Example: IN = 0 111 1111 1111 1111 1111 1111 1111 1100 └──┬──┘ └──┬──┘ SIGN OUT32...OUT1</p>
Inputs	Input (IN): DINT
Outputs	Output (OUT1...OUT32): Boolean Sign output (SIGN): Boolean

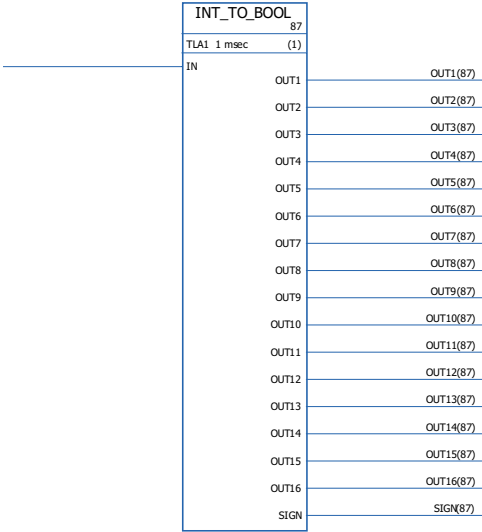
DINT_TO_INT	
(10021)	
Illustration	

Execution time	0.53 μ s								
Operation	<p>The output (O) value is a 16-bit integer value of the 32-bit integer input (I) value.</p> <p>Examples:</p> <table border="1"> <thead> <tr> <th>I (31 bits + sign)</th> <th>O (15 bits + sign)</th> </tr> </thead> <tbody> <tr> <td>2147483647</td> <td>32767</td> </tr> <tr> <td>-2147483648</td> <td>-32767</td> </tr> <tr> <td>0</td> <td>0</td> </tr> </tbody> </table>	I (31 bits + sign)	O (15 bits + sign)	2147483647	32767	-2147483648	-32767	0	0
I (31 bits + sign)	O (15 bits + sign)								
2147483647	32767								
-2147483648	-32767								
0	0								
Inputs	Input (I): DINT								
Outputs	Output (O): INT								

DINT_TO_REALn	
(10023)	
Illustration	
Execution time	7.25 μ s
Operation	<p>The output (OUT) is the REAL/REAL24 equivalent of the input (IN). Input IN1 is the integer value and input IN2 is the fractional value.</p> <p>If one (or both) of the input values is negative, the output value is negative.</p> <p>Example (from DINT to REAL):</p> <p>When IN1 = 2 and IN2 = 3276, OUT = 2.04999.</p> <p>The output value is limited to the maximum value of the selected data type range.</p>
Inputs	Input (IN1, IN2): DINT
Outputs	<p>The output data type is selected by the user.</p> <p>Output (OUT): REAL, REAL24</p>

DINT_TO_REALn_SIMP	
(10022)	
Illustration	
Execution time	6.53 μ s

<p>Operation</p>	<p>The output (O) is the REAL/REAL24 equivalent of the input (I) divided by the scale input (SCALE).</p> <p>Error codes indicated at the error output (ERRC) are as follows:</p> <table border="1" data-bbox="384 338 1286 725"> <thead> <tr> <th>Error code</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>No error</td> </tr> <tr> <td>1001</td> <td>The calculated REAL/REAL24 value exceeds the minimum value of the selected data type range. The output is set to the minimum value.</td> </tr> <tr> <td>1002</td> <td>The calculated REAL/REAL24 value exceeds the maximum value of the selected data type range. The output is set to the maximum value.</td> </tr> <tr> <td>1003</td> <td>The SCALE input is 0. The output is set to 0.</td> </tr> <tr> <td>1004</td> <td>Incorrect SCALE input, i.e. the scale input is < 0 or is not a factor of 10.</td> </tr> </tbody> </table> <p>Example (from DINT to REAL24): When I = 205 and SCALE = 100, I/SCALE = 205 /100 = 2.05 and O = 2.04999.</p>	Error code	Description	0	No error	1001	The calculated REAL/REAL24 value exceeds the minimum value of the selected data type range. The output is set to the minimum value.	1002	The calculated REAL/REAL24 value exceeds the maximum value of the selected data type range. The output is set to the maximum value.	1003	The SCALE input is 0. The output is set to 0.	1004	Incorrect SCALE input, i.e. the scale input is < 0 or is not a factor of 10.
Error code	Description												
0	No error												
1001	The calculated REAL/REAL24 value exceeds the minimum value of the selected data type range. The output is set to the minimum value.												
1002	The calculated REAL/REAL24 value exceeds the maximum value of the selected data type range. The output is set to the maximum value.												
1003	The SCALE input is 0. The output is set to 0.												
1004	Incorrect SCALE input, i.e. the scale input is < 0 or is not a factor of 10.												
<p>Inputs</p>	<p>Input (I): DINT Scale input (SCALE): DINT</p>												
<p>Outputs</p>	<p>The output data type is selected by the user. Output (O): REAL, REAL24 Error output (ERRC): DINT</p>												

<p>INT_TO_BOOL (10024)</p>	
<p>Illustration</p>	
<p>Execution time</p>	<p>4.31 µs</p>
<p>Operation</p>	<p>The boolean output (OUT1...OUT16) values are formed from the 16-bit integer input (IN) value.</p> <p>Example:</p> <p>IN = 0111 1111 1111 1111</p> <p style="margin-left: 40px;"> OUT16, A, 110U SIGN </p>

Inputs	Input (IN): INT
Outputs	Output (OUT1...OUT16): Boolean Sign output (SIGN): Boolean

INT_TO_DINT									
(10025)									
Illustration	<p>The diagram shows a rectangular block labeled 'INT_TO_DINT' with '88' in the top right corner. Inside the block, 'TLA1 1 msec (1)' is written. An input line labeled 'I' enters from the left, and an output line labeled 'O(88)' exits to the right.</p>								
Execution time	0.33 μs								
Operation	<p>The output (O) value is a 32-bit integer value of the 16-bit integer input (I) value.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>I</th> <th>O</th> </tr> </thead> <tbody> <tr> <td>32767</td> <td>32767</td> </tr> <tr> <td>-32767</td> <td>-32767</td> </tr> <tr> <td>0</td> <td>0</td> </tr> </tbody> </table>	I	O	32767	32767	-32767	-32767	0	0
I	O								
32767	32767								
-32767	-32767								
0	0								
Inputs	Input (I): INT								
Outputs	Output (O): DINT								

REAL_TO_REAL24	
(10026)	
Illustration	<p>The diagram shows a rectangular block labeled 'REAL_TO_REAL24' with '89' in the top right corner. Inside the block, 'TLA1 1 msec (1)' is written. An input line labeled 'I' enters from the left, and an output line labeled 'O(89)' exits to the right.</p>
Execution time	1.35 μs
Operation	<p>Output (O) is the REAL24 equivalent of the REAL input (I). The output value is limited to the maximum value of the data type. Example:</p> <p style="margin-left: 40px;"> I = 0000 0000 0010 0110 1111 1111 1111 1111 Integer value Fractional value </p> <p style="margin-left: 40px;"> O = 0010 0110 1111 1111 1111 1111 0000 0000 Integer value Fractional value </p>
Inputs	Input (I): REAL
Outputs	Output (O): REAL24

REAL24_TO_REAL	
(10027)	
Illustration	
Execution time	1.20 μs
Operation	<p>Output (O) is the REAL equivalent of the REAL24 input (I). The output value is limited to the maximum value of the data type range. Example:</p> <p style="margin-left: 40px;"> $I = \underbrace{0010\ 0110}_{\text{Integer value}}\ \underbrace{1111\ 1111\ 1111\ 1111\ 0000\ 0000}_{\text{Fractional value}}$ </p> <p style="margin-left: 40px;"> $O = \underbrace{0000\ 0000\ 0010\ 0110}_{\text{Integer value}}\ \underbrace{1111\ 1111\ 1111\ 1111}_{\text{Fractional value}}$ </p>
Inputs	Input (I): REAL24
Outputs	Output (O): REAL

REALn_TO_DINT	
(10029)	
Illustration	
Execution time	6.45 μs
Operation	<p>Output (O) is the 32-bit integer equivalent of the REAL/REAL24 input (I). Output O1 is the integer value and output O2 is the fractional value. The output value is limited to the maximum value of the data type range. Example (from REAL to DINT): When I = 2.04998779297, O1 = 2 and O2 = 3276.</p>
Inputs	The input data type is selected by the user. Input (I): REAL, REAL24
Outputs	Output (O1, O2): DINT

REALn_TO_DINT_SIMP	
(10028)	
Illustration	
Execution time	5.54 μs

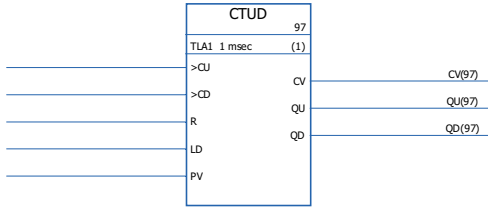
Operation	<p>Output (O) is the 32-bit integer equivalent of the REAL/REAL24 input (I) multiplied by the scale input (SCALE).</p> <p>Error codes are indicated by the error output (ERRC) as follows:</p> <table border="1" data-bbox="536 336 1310 663"> <thead> <tr> <th data-bbox="536 336 711 376">Error code</th> <th data-bbox="711 336 1310 376">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="536 376 711 416">0</td> <td data-bbox="711 376 1310 416">No error</td> </tr> <tr> <td data-bbox="536 416 711 486">1001</td> <td data-bbox="711 416 1310 486">The calculated integer value exceeds the minimum value. The output is set to the minimum value.</td> </tr> <tr> <td data-bbox="536 486 711 555">1002</td> <td data-bbox="711 486 1310 555">The calculated integer value exceeds the maximum value. The output is set to the maximum value.</td> </tr> <tr> <td data-bbox="536 555 711 595">1003</td> <td data-bbox="711 555 1310 595">Scale input is 0. The output is set to 0.</td> </tr> <tr> <td data-bbox="536 595 711 663">1004</td> <td data-bbox="711 595 1310 663">Incorrect scale input, i.e. scale input is < 0 or is not a factor of 10.</td> </tr> </tbody> </table> <p>Example (from REAL to DINT): When I = 2.04998779297 and SCALE = 100, O = 204.</p>	Error code	Description	0	No error	1001	The calculated integer value exceeds the minimum value. The output is set to the minimum value.	1002	The calculated integer value exceeds the maximum value. The output is set to the maximum value.	1003	Scale input is 0. The output is set to 0.	1004	Incorrect scale input, i.e. scale input is < 0 or is not a factor of 10.
Error code	Description												
0	No error												
1001	The calculated integer value exceeds the minimum value. The output is set to the minimum value.												
1002	The calculated integer value exceeds the maximum value. The output is set to the maximum value.												
1003	Scale input is 0. The output is set to 0.												
1004	Incorrect scale input, i.e. scale input is < 0 or is not a factor of 10.												
Inputs	<p>The input data type is selected by the user.</p> <p>Input (I): REAL, REAL24</p> <p>Scale input (SCALE): DINT</p>												
Outputs	<p>Output (O): DINT</p> <p>Error output (ERRC): DINT</p>												

Counters

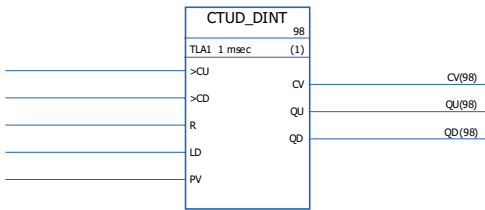
CTD (10047)																																																	
Illustration																																																	
Execution time	0.92 μs																																																
Operation	<p>The counter output (CV) value is decreased by 1 if the counter input (CD) value changes from 0 -> 1 and the load input (LD) value is 0. If the load input value is 1, the preset input (PV) value is stored as the counter output (CV) value. If the counter output has reached its minimum value -32768, the counter output remains unchanged.</p> <p>The status output (Q) is 1 if the counter output (CV) value ≤ 0.</p> <p>Example:</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>LD</th> <th>CD</th> <th>PV</th> <th>Q</th> <th>CV_{prev}</th> <th>CV</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1 -> 0</td> <td>10</td> <td>0</td> <td>5</td> <td>5</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>10</td> <td>0</td> <td>5</td> <td>5 - 1 = 4</td> </tr> <tr> <td>1</td> <td>1 -> 0</td> <td>-2</td> <td>1</td> <td>4</td> <td>-2</td> </tr> <tr> <td>1</td> <td>0 -> 1</td> <td>1</td> <td>0</td> <td>-2</td> <td>1</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>5</td> <td>1</td> <td>1</td> <td>1 - 1 = 0</td> </tr> <tr> <td>1</td> <td>1 -> 0</td> <td>-32768</td> <td>1</td> <td>0</td> <td>-32768</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>10</td> <td>1</td> <td>-32768</td> <td>-32768</td> </tr> </tbody> </table> <p>CV_{prev} is the previous cycle counter output value.</p>	LD	CD	PV	Q	CV _{prev}	CV	0	1 -> 0	10	0	5	5	0	0 -> 1	10	0	5	5 - 1 = 4	1	1 -> 0	-2	1	4	-2	1	0 -> 1	1	0	-2	1	0	0 -> 1	5	1	1	1 - 1 = 0	1	1 -> 0	-32768	1	0	-32768	0	0 -> 1	10	1	-32768	-32768
LD	CD	PV	Q	CV _{prev}	CV																																												
0	1 -> 0	10	0	5	5																																												
0	0 -> 1	10	0	5	5 - 1 = 4																																												
1	1 -> 0	-2	1	4	-2																																												
1	0 -> 1	1	0	-2	1																																												
0	0 -> 1	5	1	1	1 - 1 = 0																																												
1	1 -> 0	-32768	1	0	-32768																																												
0	0 -> 1	10	1	-32768	-32768																																												
Inputs	Load input (LD): Boolean Counter input (CD): Boolean Preset input (PV): INT																																																
Outputs	Counter output (CV): INT Status output (Q): Boolean																																																

CTD_DINT (10046)	
Illustration	
Execution time	0.92 μs

<p>Operation</p>	<p>The counter output (CV) value is decreased by 1 if the counter input (CD) value changes from 0 -> 1 and the load input (LD) value is 0. If the load input (LD) value is 1, the preset input (PV) value is stored as the counter output (CV) value. If the counter output has reached its minimum value -2147483648, the counter output remains unchanged.</p> <p>The status output (Q) is 1 if the counter output (CV) value < 0.</p> <p>Example:</p> <table border="1" data-bbox="544 459 1437 835"> <thead> <tr> <th>LD</th> <th>CD</th> <th>PV</th> <th>Q</th> <th>CV_{prev}</th> <th>CV</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1 -> 0</td> <td>10</td> <td>0</td> <td>5</td> <td>5</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>10</td> <td>0</td> <td>5</td> <td>5 - 1 = 4</td> </tr> <tr> <td>1</td> <td>1 -> 0</td> <td>-2</td> <td>1</td> <td>4</td> <td>-2</td> </tr> <tr> <td>1</td> <td>0 -> 1</td> <td>1</td> <td>0</td> <td>-2</td> <td>1</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>5</td> <td>1</td> <td>1</td> <td>1 - 1 = 0</td> </tr> <tr> <td>1</td> <td>1 -> 0</td> <td>-2147483648</td> <td>1</td> <td>0</td> <td>-2147483648</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>10</td> <td>1</td> <td>-2147483648</td> <td>-2147483648</td> </tr> </tbody> </table> <p>CV_{prev} is the previous cycle counter output value.</p>	LD	CD	PV	Q	CV _{prev}	CV	0	1 -> 0	10	0	5	5	0	0 -> 1	10	0	5	5 - 1 = 4	1	1 -> 0	-2	1	4	-2	1	0 -> 1	1	0	-2	1	0	0 -> 1	5	1	1	1 - 1 = 0	1	1 -> 0	-2147483648	1	0	-2147483648	0	0 -> 1	10	1	-2147483648	-2147483648
LD	CD	PV	Q	CV _{prev}	CV																																												
0	1 -> 0	10	0	5	5																																												
0	0 -> 1	10	0	5	5 - 1 = 4																																												
1	1 -> 0	-2	1	4	-2																																												
1	0 -> 1	1	0	-2	1																																												
0	0 -> 1	5	1	1	1 - 1 = 0																																												
1	1 -> 0	-2147483648	1	0	-2147483648																																												
0	0 -> 1	10	1	-2147483648	-2147483648																																												
<p>Inputs</p>	<p>Load input (LD): Boolean Counter input (CD): Boolean Preset input (PV): DINT</p>																																																
<p>Outputs</p>	<p>Counter output (CV): DINT Status output (Q): Boolean</p>																																																

<p>CTU (10049)</p>	
<p>Illustration</p>	 <p>The diagram shows a CTUD (97) block with a timer T1A1 1 msec (1). On the left, there are five input lines: >CU, >CD, R, LD, and PV. On the right, there are three output lines: CV(97), QU(97), and QD(97).</p>
<p>Execution time</p>	<p>0.92 μs</p>

<p>Operation</p>	<p>The counter output (CV) value is increased by 1 if the counter input (CU) value changes from 0 -> 1 and the reset input (R) value is 0. If the counter output has reached its maximum value 32767, the counter output remains unchanged.</p> <p>The counter output (CV) is reset to 0 if the reset input (R) is 1.</p> <p>The status output (Q) is 1 if the counter output (CV) value \geq preset input (PV) value.</p> <p>Example:</p> <table border="1" data-bbox="384 439 1286 719"> <thead> <tr> <th>R</th> <th>CU</th> <th>PV</th> <th>Q</th> <th>CV_{prev}</th> <th>CV</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1 -> 0</td> <td>20</td> <td>0</td> <td>10</td> <td>10</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>11</td> <td>1</td> <td>10</td> <td>10 + 1 = 11</td> </tr> <tr> <td>1</td> <td>1 -> 0</td> <td>20</td> <td>0</td> <td>11</td> <td>0</td> </tr> <tr> <td>1</td> <td>0 -> 1</td> <td>5</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>20</td> <td>0</td> <td>0</td> <td>0 + 1 = 1</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>30</td> <td>1</td> <td>32767</td> <td>32767</td> </tr> </tbody> </table> <p>CV_{prev} is the previous cycle counter output value.</p>	R	CU	PV	Q	CV _{prev}	CV	0	1 -> 0	20	0	10	10	0	0 -> 1	11	1	10	10 + 1 = 11	1	1 -> 0	20	0	11	0	1	0 -> 1	5	0	0	0	0	0 -> 1	20	0	0	0 + 1 = 1	0	0 -> 1	30	1	32767	32767
R	CU	PV	Q	CV _{prev}	CV																																						
0	1 -> 0	20	0	10	10																																						
0	0 -> 1	11	1	10	10 + 1 = 11																																						
1	1 -> 0	20	0	11	0																																						
1	0 -> 1	5	0	0	0																																						
0	0 -> 1	20	0	0	0 + 1 = 1																																						
0	0 -> 1	30	1	32767	32767																																						
<p>Inputs</p>	<p>Counter input (CU): Boolean Reset input (R): Boolean Preset input (PV): INT</p>																																										
<p>Outputs</p>	<p>Counter output (CV): INT Status output (Q): Boolean</p>																																										

<p>CTU_DINT (10048)</p>																																											
<p>Illustration</p>																																											
<p>Execution time</p>	<p>0.92 μs</p>																																										
<p>Operation</p>	<p>The counter output (CV) value is increased by 1 if the counter input (CU) value changes from 0 -> 1 and the reset input (R) value is 0. If the counter output has reached its maximum value 2147483647, the counter output remains unchanged.</p> <p>The counter output (CV) is reset to 0 if the reset input (R) is 1.</p> <p>The status output (Q) is 1 if the counter output (CV) value \geq preset input (PV) value.</p> <p>Example:</p> <table border="1" data-bbox="384 1659 1286 1939"> <thead> <tr> <th>R</th> <th>CU</th> <th>PV</th> <th>Q</th> <th>CV_{prev}</th> <th>CV</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1 -> 0</td> <td>20</td> <td>0</td> <td>10</td> <td>10</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>11</td> <td>1</td> <td>10</td> <td>10 + 1 = 11</td> </tr> <tr> <td>1</td> <td>1 -> 0</td> <td>20</td> <td>0</td> <td>11</td> <td>0</td> </tr> <tr> <td>1</td> <td>0 -> 1</td> <td>5</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>20</td> <td>0</td> <td>0</td> <td>0 + 1 = 1</td> </tr> <tr> <td>0</td> <td>0 -> 1</td> <td>30</td> <td>1</td> <td>2147483647</td> <td>2147483647</td> </tr> </tbody> </table> <p>CV_{prev} is the previous cycle counter output value.</p>	R	CU	PV	Q	CV _{prev}	CV	0	1 -> 0	20	0	10	10	0	0 -> 1	11	1	10	10 + 1 = 11	1	1 -> 0	20	0	11	0	1	0 -> 1	5	0	0	0	0	0 -> 1	20	0	0	0 + 1 = 1	0	0 -> 1	30	1	2147483647	2147483647
R	CU	PV	Q	CV _{prev}	CV																																						
0	1 -> 0	20	0	10	10																																						
0	0 -> 1	11	1	10	10 + 1 = 11																																						
1	1 -> 0	20	0	11	0																																						
1	0 -> 1	5	0	0	0																																						
0	0 -> 1	20	0	0	0 + 1 = 1																																						
0	0 -> 1	30	1	2147483647	2147483647																																						

Inputs	Counter input (CU): Boolean Reset input (R): Boolean Preset input (PV): DINT
Outputs	Counter output (CV): DINT Status output (Q): Boolean

CTUD (10051)	
Illustration	
Execution time	1.40 μ s
Operation	<p>The counter output (CV) value is increased by 1 if the counter input (CU) value changes from 0 -> 1 and the reset input (R) is 0 and the load input (LD) is 0.</p> <p>The counter output (CV) value is decreased by 1 if the counter input (CD) changes from 0 -> 1 and the load input (LD) is 0 and the reset input (R) is 0.</p> <p>If the load input (LD) is 1, the preset input (PV) value is stored as the counter output (CV) value.</p> <p>The counter output (CV) is reset to 0 if the reset input (R) is 1.</p> <p>If the counter output has reached its minimum or maximum value, -32768 or +32767, the counter output remains unchanged until it is reset (R) or until the load input (LD) is set to 1.</p> <p>The up counter status output (QU) is 1 if the counter output (CV) value \geq preset input (PV) value.</p> <p>The down counter status output (QD) is 1 if the counter output (CV) value \leq 0.</p>

	<p>Example:</p> <table border="1"> <thead> <tr> <th>CU</th> <th>CD</th> <th>R</th> <th>LD</th> <th>PV</th> <th>QU</th> <th>QD</th> <th>CV_{prev}</th> <th>CV</th> </tr> </thead> <tbody> <tr><td>0 -> 0</td><td>0 -> 0</td><td>0</td><td>0</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0 -> 0</td><td>0 -> 0</td><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td><td>0</td><td>2</td></tr> <tr><td>0 -> 0</td><td>0 -> 0</td><td>1</td><td>0</td><td>2</td><td>0</td><td>1</td><td>2</td><td>0</td></tr> <tr><td>0 -> 0</td><td>0 -> 0</td><td>1</td><td>1</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0 -> 0</td><td>0 -> 1</td><td>0</td><td>0</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0 - 1 = -1</td></tr> <tr><td>0 -> 0</td><td>1 -> 1</td><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td><td>-1</td><td>2</td></tr> <tr><td>0 -> 0</td><td>1 -> 1</td><td>1</td><td>0</td><td>2</td><td>0</td><td>1</td><td>2</td><td>0</td></tr> <tr><td>0 -> 0</td><td>1 -> 1</td><td>1</td><td>1</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0 -> 1</td><td>1 -> 0</td><td>0</td><td>0</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0 + 1 = 1</td></tr> <tr><td>1 -> 1</td><td>0 -> 0</td><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td><td>1</td><td>2</td></tr> <tr><td>1 -> 1</td><td>0 -> 0</td><td>1</td><td>0</td><td>2</td><td>0</td><td>1</td><td>2</td><td>0</td></tr> <tr><td>1 -> 1</td><td>0 -> 0</td><td>1</td><td>1</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>1 -> 1</td><td>0 -> 1</td><td>0</td><td>0</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0 - 1 = -1</td></tr> <tr><td>1 -> 1</td><td>1 -> 1</td><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td><td>-1</td><td>2</td></tr> <tr><td>1 -> 1</td><td>1 -> 1</td><td>1</td><td>0</td><td>2</td><td>0</td><td>1</td><td>2</td><td>0</td></tr> <tr><td>1 -> 1</td><td>1 -> 1</td><td>1</td><td>1</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> </tbody> </table> <p>CV_{prev} is the previous cycle counter output value.</p>	CU	CD	R	LD	PV	QU	QD	CV _{prev}	CV	0 -> 0	0 -> 0	0	0	2	0	1	0	0	0 -> 0	0 -> 0	0	1	2	1	0	0	2	0 -> 0	0 -> 0	1	0	2	0	1	2	0	0 -> 0	0 -> 0	1	1	2	0	1	0	0	0 -> 0	0 -> 1	0	0	2	0	1	0	0 - 1 = -1	0 -> 0	1 -> 1	0	1	2	1	0	-1	2	0 -> 0	1 -> 1	1	0	2	0	1	2	0	0 -> 0	1 -> 1	1	1	2	0	1	0	0	0 -> 1	1 -> 0	0	0	2	0	0	0	0 + 1 = 1	1 -> 1	0 -> 0	0	1	2	1	0	1	2	1 -> 1	0 -> 0	1	0	2	0	1	2	0	1 -> 1	0 -> 0	1	1	2	0	1	0	0	1 -> 1	0 -> 1	0	0	2	0	1	0	0 - 1 = -1	1 -> 1	1 -> 1	0	1	2	1	0	-1	2	1 -> 1	1 -> 1	1	0	2	0	1	2	0	1 -> 1	1 -> 1	1	1	2	0	1	0	0
CU	CD	R	LD	PV	QU	QD	CV _{prev}	CV																																																																																																																																																		
0 -> 0	0 -> 0	0	0	2	0	1	0	0																																																																																																																																																		
0 -> 0	0 -> 0	0	1	2	1	0	0	2																																																																																																																																																		
0 -> 0	0 -> 0	1	0	2	0	1	2	0																																																																																																																																																		
0 -> 0	0 -> 0	1	1	2	0	1	0	0																																																																																																																																																		
0 -> 0	0 -> 1	0	0	2	0	1	0	0 - 1 = -1																																																																																																																																																		
0 -> 0	1 -> 1	0	1	2	1	0	-1	2																																																																																																																																																		
0 -> 0	1 -> 1	1	0	2	0	1	2	0																																																																																																																																																		
0 -> 0	1 -> 1	1	1	2	0	1	0	0																																																																																																																																																		
0 -> 1	1 -> 0	0	0	2	0	0	0	0 + 1 = 1																																																																																																																																																		
1 -> 1	0 -> 0	0	1	2	1	0	1	2																																																																																																																																																		
1 -> 1	0 -> 0	1	0	2	0	1	2	0																																																																																																																																																		
1 -> 1	0 -> 0	1	1	2	0	1	0	0																																																																																																																																																		
1 -> 1	0 -> 1	0	0	2	0	1	0	0 - 1 = -1																																																																																																																																																		
1 -> 1	1 -> 1	0	1	2	1	0	-1	2																																																																																																																																																		
1 -> 1	1 -> 1	1	0	2	0	1	2	0																																																																																																																																																		
1 -> 1	1 -> 1	1	1	2	0	1	0	0																																																																																																																																																		
Inputs	<p>Up counter input (CU): Boolean Down counter input (CD): Boolean Reset input (R): Boolean Load input (LD): Boolean Preset input (PV): INT</p>																																																																																																																																																									
Outputs	<p>Counter output (CV): INT Up counter status output (QU): Boolean Down counter status output (QD): Boolean</p>																																																																																																																																																									

<p>CTUD_DINT (10051)</p>	
Illustration	
Execution time	1.40 μs

<p>Operation</p>	<p>The counter output (CV) value is increased by 1 if the counter input (CU) changes from 0 -> 1 and the reset input (R) is 0 and the load input (LD) is 0.</p> <p>The counter output (CV) value is decreased by 1 if the counter input (CD) changes from 0 -> 1 and the load input (LD) is 0 and the reset input (R) is 0.</p> <p>If the counter output has reached its minimum or maximum value, -2147483648 or +2147483647, the counter output remains unchanged until it is reset (R) or until the load input (LD) is set to 1.</p> <p>If the load input (LD) value is 1, the preset input (PV) value is stored as the counter output (CV) value.</p> <p>The counter output (CV) is reset to 0 if the reset input (R) is 1.</p> <p>The up counter status output (QU) is 1 if the counter output (CV) value \geq preset input (PV) value.</p> <p>The down counter status output (QD) is 1 if the counter output (CV) value \leq 0.</p> <p>Example:</p> <table border="1" data-bbox="539 703 1439 1384"> <thead> <tr> <th>CU</th> <th>CD</th> <th>R</th> <th>LD</th> <th>PV</th> <th>QU</th> <th>QD</th> <th>CV_{prev}</th> <th>CV</th> </tr> </thead> <tbody> <tr><td>0 -> 0</td><td>0 -> 0</td><td>0</td><td>0</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0 -> 0</td><td>0 -> 0</td><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td><td>0</td><td>2</td></tr> <tr><td>0 -> 0</td><td>0 -> 0</td><td>1</td><td>0</td><td>2</td><td>0</td><td>1</td><td>2</td><td>0</td></tr> <tr><td>0 -> 0</td><td>0 -> 0</td><td>1</td><td>1</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0 -> 0</td><td>0 -> 1</td><td>0</td><td>0</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0 - 1 = -1</td></tr> <tr><td>0 -> 0</td><td>1 -> 1</td><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td><td>-1</td><td>2</td></tr> <tr><td>0 -> 0</td><td>1 -> 1</td><td>1</td><td>0</td><td>2</td><td>0</td><td>1</td><td>2</td><td>0</td></tr> <tr><td>0 -> 0</td><td>1 -> 1</td><td>1</td><td>1</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0 -> 1</td><td>1 -> 0</td><td>0</td><td>0</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0 + 1 = 1</td></tr> <tr><td>1 -> 1</td><td>0 -> 0</td><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td><td>1</td><td>2</td></tr> <tr><td>1 -> 1</td><td>0 -> 0</td><td>1</td><td>0</td><td>2</td><td>0</td><td>1</td><td>2</td><td>0</td></tr> <tr><td>1 -> 1</td><td>0 -> 0</td><td>1</td><td>1</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>1 -> 1</td><td>0 -> 1</td><td>0</td><td>0</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0 - 1 = -1</td></tr> <tr><td>1 -> 1</td><td>1 -> 1</td><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td><td>-1</td><td>2</td></tr> <tr><td>1 -> 1</td><td>1 -> 1</td><td>1</td><td>0</td><td>2</td><td>0</td><td>1</td><td>2</td><td>0</td></tr> <tr><td>1 -> 1</td><td>1 -> 1</td><td>1</td><td>1</td><td>2</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> </tbody> </table> <p>CV_{prev} is the previous cycle counter output value.</p>	CU	CD	R	LD	PV	QU	QD	CV _{prev}	CV	0 -> 0	0 -> 0	0	0	2	0	1	0	0	0 -> 0	0 -> 0	0	1	2	1	0	0	2	0 -> 0	0 -> 0	1	0	2	0	1	2	0	0 -> 0	0 -> 0	1	1	2	0	1	0	0	0 -> 0	0 -> 1	0	0	2	0	1	0	0 - 1 = -1	0 -> 0	1 -> 1	0	1	2	1	0	-1	2	0 -> 0	1 -> 1	1	0	2	0	1	2	0	0 -> 0	1 -> 1	1	1	2	0	1	0	0	0 -> 1	1 -> 0	0	0	2	0	0	0	0 + 1 = 1	1 -> 1	0 -> 0	0	1	2	1	0	1	2	1 -> 1	0 -> 0	1	0	2	0	1	2	0	1 -> 1	0 -> 0	1	1	2	0	1	0	0	1 -> 1	0 -> 1	0	0	2	0	1	0	0 - 1 = -1	1 -> 1	1 -> 1	0	1	2	1	0	-1	2	1 -> 1	1 -> 1	1	0	2	0	1	2	0	1 -> 1	1 -> 1	1	1	2	0	1	0	0
CU	CD	R	LD	PV	QU	QD	CV _{prev}	CV																																																																																																																																																		
0 -> 0	0 -> 0	0	0	2	0	1	0	0																																																																																																																																																		
0 -> 0	0 -> 0	0	1	2	1	0	0	2																																																																																																																																																		
0 -> 0	0 -> 0	1	0	2	0	1	2	0																																																																																																																																																		
0 -> 0	0 -> 0	1	1	2	0	1	0	0																																																																																																																																																		
0 -> 0	0 -> 1	0	0	2	0	1	0	0 - 1 = -1																																																																																																																																																		
0 -> 0	1 -> 1	0	1	2	1	0	-1	2																																																																																																																																																		
0 -> 0	1 -> 1	1	0	2	0	1	2	0																																																																																																																																																		
0 -> 0	1 -> 1	1	1	2	0	1	0	0																																																																																																																																																		
0 -> 1	1 -> 0	0	0	2	0	0	0	0 + 1 = 1																																																																																																																																																		
1 -> 1	0 -> 0	0	1	2	1	0	1	2																																																																																																																																																		
1 -> 1	0 -> 0	1	0	2	0	1	2	0																																																																																																																																																		
1 -> 1	0 -> 0	1	1	2	0	1	0	0																																																																																																																																																		
1 -> 1	0 -> 1	0	0	2	0	1	0	0 - 1 = -1																																																																																																																																																		
1 -> 1	1 -> 1	0	1	2	1	0	-1	2																																																																																																																																																		
1 -> 1	1 -> 1	1	0	2	0	1	2	0																																																																																																																																																		
1 -> 1	1 -> 1	1	1	2	0	1	0	0																																																																																																																																																		
<p>Inputs</p>	<p>Up counter input (CU): Boolean Down counter input (CD): Boolean Reset input (R): Boolean Load input (LD): Boolean Preset input (PV): DINT</p>																																																																																																																																																									
<p>Outputs</p>	<p>Counter output (CV): DINT Up counter status output (QU): Boolean Down counter status output (QD): Boolean</p>																																																																																																																																																									

Edge & bistable

FTRIG																
(10030)																
Illustration																
Execution time	0.38 μs															
Operation	<p>The output (Q) is set to 1 when the clock input (CLK) changes from 1 to 0. The output is set back to 0 with the next execution of the block. Otherwise the output is 0.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>CLK_{previous}</th> <th>CLK</th> <th>Q</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1 (for one execution cycle time, returns to 0 at the next execution)</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table> <p>CLK_{previous} is the previous cycle output value.</p>	CLK _{previous}	CLK	Q	0	0	0	0	1	0	1	0	1 (for one execution cycle time, returns to 0 at the next execution)	1	1	0
CLK _{previous}	CLK	Q														
0	0	0														
0	1	0														
1	0	1 (for one execution cycle time, returns to 0 at the next execution)														
1	1	0														
Inputs	Clock input (CLK): Boolean															
Outputs	Output (Q): Boolean															

RS																																					
(10032)																																					
Illustration																																					
Execution time	0.38 μs																																				
Operation	<p>The output (Q1) is 1 if the set input (S) is 1 and the reset input (R1) is 0. The output will retain the previous output state if the set input (S) and the reset input (R1) are 0. The output is 0 if the reset input is 1.</p> <p>Truth table:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>S</th> <th>R1</th> <th>Q1_{previous}</th> <th>Q1</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table> <p>Q_{previous} is the previous cycle output value.</p>	S	R1	Q1 _{previous}	Q1	0	0	0	0	0	0	1	1	0	1	0	0	0	1	1	0	1	0	0	1	1	0	1	1	1	1	0	0	1	1	1	0
S	R1	Q1 _{previous}	Q1																																		
0	0	0	0																																		
0	0	1	1																																		
0	1	0	0																																		
0	1	1	0																																		
1	0	0	1																																		
1	0	1	1																																		
1	1	0	0																																		
1	1	1	0																																		

Inputs	Set input (S): Boolean Reset input (R1): Boolean
Outputs	Output (Q1): Boolean

RTRIG (10031)																
Illustration																
Execution time	0.38 μs															
Operation	<p>The output (Q) is set to 1 when the clock input (CLK) changes from 0 to 1. The output is set back to 0 with the next execution of the block. Otherwise the output is 0.</p> <table border="1"> <thead> <tr> <th>CLK_{previous}</th> <th>CLK</th> <th>Q</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table> <p>CLK_{previous} is the previous cycle output value.</p> <p>Note: The output (Q) is 1 after the first execution of the block after cold restart when the clock input (CLK) is 1. Otherwise the output is always 0 when the clock input is 1.</p>	CLK _{previous}	CLK	Q	0	0	0	0	1	1	1	0	0	1	1	0
CLK _{previous}	CLK	Q														
0	0	0														
0	1	1														
1	0	0														
1	1	0														
Inputs	Clock input (CLK): Boolean															
Outputs	Output (Q): Boolean															

SR (10033)	
Illustration	
Execution time	0.38 μs

54 Standard function blocks

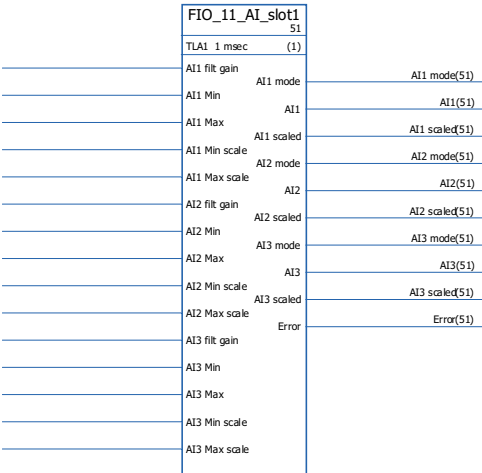
<p>Operation</p>	<p>The output (Q1) is 1 if the set input (S1) is 1. The output will retain the previous output state if the set input (S1) and the reset input (R) are 0. The output is 0 if the set input is 0 and the reset input is 1.</p> <p>Truth table:</p> <table border="1" data-bbox="379 367 1054 725"> <thead> <tr> <th>S1</th> <th>R</th> <th>Q1_{previous}</th> <th>Q1</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table> <p>Q1_{previous} is the previous cycle output value.</p>	S1	R	Q1 _{previous}	Q1	0	0	0	0	0	0	1	1	0	1	0	0	0	1	1	0	1	0	0	1	1	0	1	1	1	1	0	1	1	1	1	1
S1	R	Q1 _{previous}	Q1																																		
0	0	0	0																																		
0	0	1	1																																		
0	1	0	0																																		
0	1	1	0																																		
1	0	0	1																																		
1	0	1	1																																		
1	1	0	1																																		
1	1	1	1																																		
<p>Inputs</p>	<p>Set input (S1): Boolean Reset input (R): Boolean</p>																																				
<p>Outputs</p>	<p>Output (Q1): Boolean</p>																																				

Extensions

FIO_01_slot1 (10084)	
Illustration	<p>The diagram shows the FIO_01_slot1 block with the following connections:</p> <ul style="list-style-type: none"> Inputs: TLA1 1 msec (1), DIO1 conf, DIO2 conf, DIO3 conf, DIO4 conf, DO1, DO2, DO3, DO4, RO1, RO2. Outputs: DI1(49), DI2(49), DI3(49), DI4(49), Error(49).
Execution time	8.6 μ s
Operation	<p>The block controls the four digital inputs/outputs (DIO1...DIO4) and two relay outputs (RO1, RO2) of a FIO-01 Digital I/O Extension mounted on slot 1 of the drive control unit.</p> <p>The state of a DIOx conf input of the block determines whether the corresponding DIO on the FIO-01 is an input or an output (0 = input, 1 = output). If the DIO is an output, the DOx input of the block defines its state.</p> <p>The RO1 and RO2 inputs define the state of the relay outputs of the FIO-01 (0 = not energized, 1 = energized).</p> <p>The DIx outputs show the state of the DIOs.</p>
Inputs	<p>Digital input/output mode selection (DIO1 conf ... DIO4 conf): Boolean</p> <p>Digital output state selection (DO1...DO4): Boolean</p> <p>Relay output state selection (RO1, RO2): Boolean</p>
Outputs	<p>Digital input/output state (DI1...DI4): Boolean</p> <p>Error output (Error): DINT (0 = No error; 1 = Application program memory full)</p>

FIO_01_slot2 (10085)	
Illustration	<p>The diagram shows the FIO_01_slot2 block with the following connections:</p> <ul style="list-style-type: none"> Inputs: TLA1 1 msec (1), DIO1 conf, DIO2 conf, DIO3 conf, DIO4 conf, DO1, DO2, DO3, DO4, RO1, RO2. Outputs: DI1(50), DI2(50), DI3(50), DI4(50), Error(50).
Execution time	8.6 μ s

Operation	<p>The block controls the four digital inputs/outputs (DIO1...DIO4) and two relay outputs (RO1, RO2) of a FIO-01 Digital I/O Extension mounted on slot 2 of the drive control unit.</p> <p>The state of a DIOx conf input of the block determines whether the corresponding DIO on the FIO-01 is an input or an output (0 = input, 1 = output). If the DIO is an output, the DOx input of the block defines its state.</p> <p>The RO1 and RO2 inputs define the state of the relay outputs of the FIO-01 (0 = not energised, 1 = energised).</p> <p>The DIx outputs show the state of the DIOs.</p>
Inputs	<p>Digital input/output mode selection (DIO1 conf ... DIO4 conf): Boolean</p> <p>Digital output state selection (DO1...DO4): Boolean</p> <p>Relay output state selection (RO1, RO2): Boolean</p>
Outputs	<p>Digital input/output state (DI1...DI4): Boolean</p> <p>Error output (Error): DINT (0 = No error; 1 = Application program memory full)</p>

<h2 style="margin: 0;">FIO_11_AI_slot1</h2> <p style="margin: 0;">(10088)</p>	
<p>Illustration</p>	 <p>The diagram shows the internal structure of the FIO_11_AI_slot1 block. It features a central vertical box with the following elements from top to bottom:</p> <ul style="list-style-type: none"> TLAI 1 msec (1) AI1 mode AI1 scaled AI1 AI1 mode(S1) AI1(S1) AI1 scaled(S1) AI2 mode AI2 mode(S1) AI2 AI2(S1) AI2 scaled(S1) AI3 mode AI3 mode(S1) AI3(S1) AI3 scaled(S1) Error Error(S1) AI3 filter gain AI3 Min AI3 Max AI3 Min scale AI3 Max scale <p>Horizontal lines connect these internal components to external inputs and outputs on the left and right sides of the block.</p>
<p>Execution time</p>	<p>11.1 µs</p>

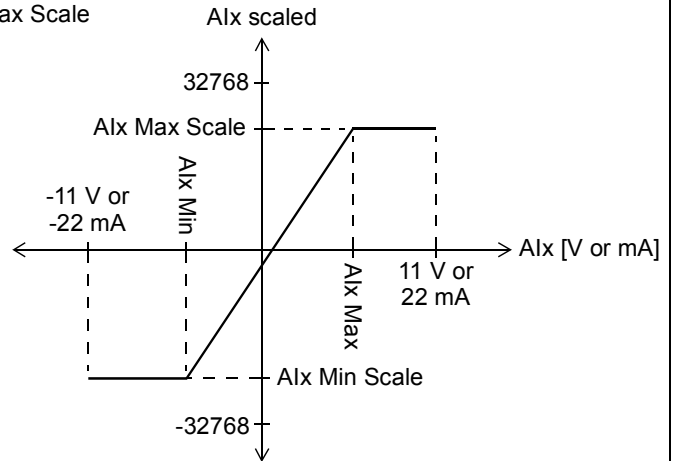
Operation

The block controls the three analogue inputs (AI1...AI3) of a FIO-11 Analog I/O Extension mounted on slot 1 of the drive control unit.

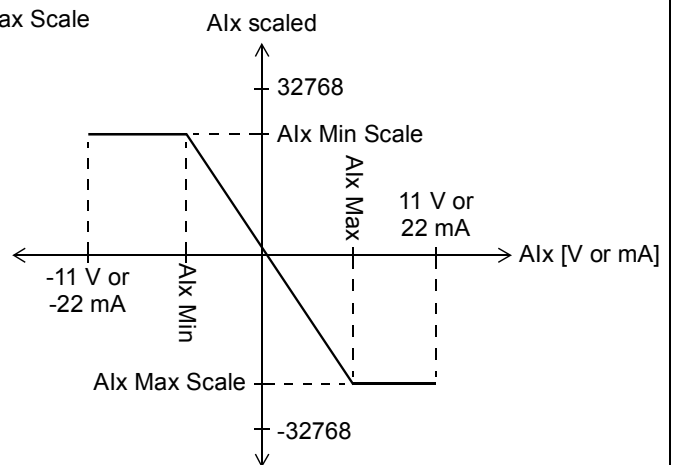
The block outputs both the unscaled (Alx) and scaled (Alx scaled) actual values of each analogue input. The scaling is based on the relationship between the ranges Alx min ... Alx max and Alx min scale ... Alx max scale.

Alx Min must be smaller than Alx Max; Alx Max Scale can be greater or smaller than Alx Min Scale.

Alx Min Scale < Alx Max Scale



Alx Min Scale > Alx Max Scale



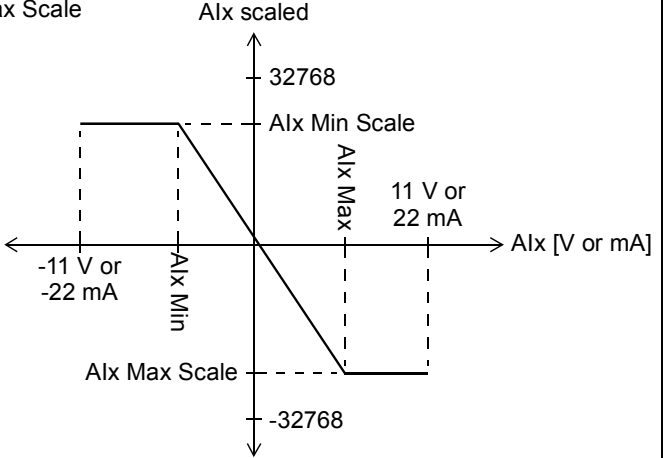
The Alx filt gain inputs determine a filtering time for each input as follows:

Alx filt gain	Filtering time	Notes
0	No filtering	
1	125 μ s	Recommended setting
2	250 μ s	
3	500 μ s	
4	1 ms	
5	2 ms	
6	4 ms	
7	7.9375 ms	

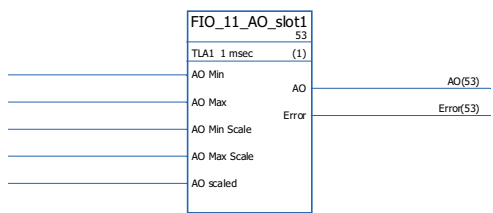
The Alx mode outputs show whether the corresponding input is voltage (0) or current (1). The voltage/current selection is made using the hardware switches on the FIO-11.

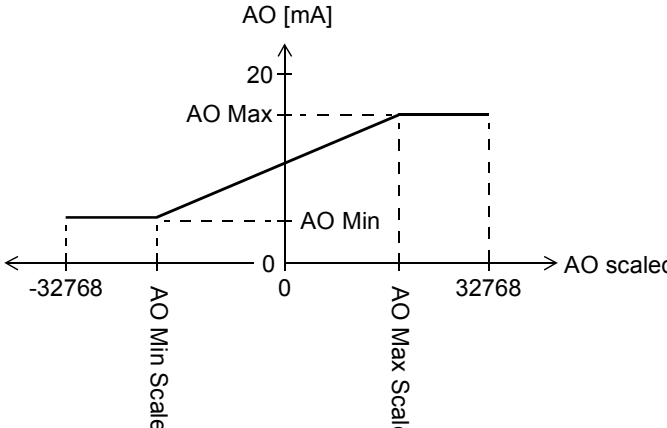
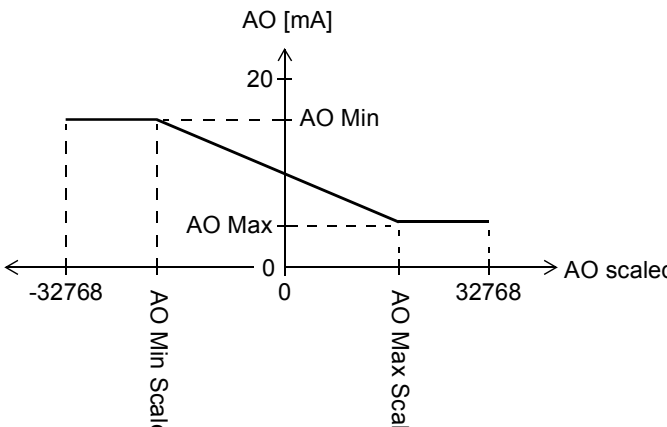
Inputs	Analogue input filter gain selection (AI1 filt gain ... AI3 filt gain): INT Minimum value of input signal (AI1 Min ... AI3 Min): REAL (≥ -11 V or -22 mA) Maximum value of input signal (AI1 Max ... AI3 Max): REAL (≤ 11 V or 22 mA) Minimum value of scaled output signal (AI1 Min scale ... AI3 Min scale): REAL Maximum value of scaled output signal (AI1 Max scale ... AI3 Max scale): REAL
Outputs	Analogue input mode (voltage or current) (AI1 mode ... AI3 mode): Boolean Value of analogue input (AI1 ... AI3): REAL Scaled value of analogue input (AI1 scaled ... AI3 scaled): REAL Error output (Error): DINT (0 = No error; 1 = Application program memory full)

FIO_11_AI_slot2 (10089)	
Illustration	
Execution time	11.1 μ s
Operation	<p>The block controls the three analogue inputs (AI1...AI3) of a FIO-11 Analog I/O Extension mounted on slot 2 of the drive control unit.</p> <p>The block outputs both the unscaled (AIx) and scaled (AIx scaled) actual values of each analogue input. The scaling is based on the relationship between the ranges AIx min ... AIx max and AIx min scale ... AIx max scale.</p> <p>AIx Min must be smaller than AIx Max; AIx Max Scale can be greater or smaller than AIx Min Scale.</p> <p>AIx Min Scale < AIx Max Scale</p>

	<p>Alx Min Scale > Alx Max Scale</p>  <p>The Alx filt gain inputs determine a filtering time for each input as follows:</p> <table border="1" data-bbox="534 831 1173 1189"> <thead> <tr> <th>Alx filt gain</th> <th>Filtering time</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>No filtering</td> <td></td> </tr> <tr> <td>1</td> <td>125 μs</td> <td>Recommended setting</td> </tr> <tr> <td>2</td> <td>250 μs</td> <td></td> </tr> <tr> <td>3</td> <td>500 μs</td> <td></td> </tr> <tr> <td>4</td> <td>1 ms</td> <td></td> </tr> <tr> <td>5</td> <td>2 ms</td> <td></td> </tr> <tr> <td>6</td> <td>4 ms</td> <td></td> </tr> <tr> <td>7</td> <td>7.9375 ms</td> <td></td> </tr> </tbody> </table> <p>The Alx mode outputs show whether the corresponding input is voltage (0) or current (1). The voltage/current selection is made using the hardware switches on the FIO-11.</p>	Alx filt gain	Filtering time	Notes	0	No filtering		1	125 μ s	Recommended setting	2	250 μ s		3	500 μ s		4	1 ms		5	2 ms		6	4 ms		7	7.9375 ms	
Alx filt gain	Filtering time	Notes																										
0	No filtering																											
1	125 μ s	Recommended setting																										
2	250 μ s																											
3	500 μ s																											
4	1 ms																											
5	2 ms																											
6	4 ms																											
7	7.9375 ms																											
<p>Inputs</p>	<p>Analogue input filter gain selection (AI1 filt gain ... AI3 filt gain): INT Minimum value of input signal (AI1 Min ... AI3 Min): REAL (\geq -11 V or -22 mA) Maximum value of input signal (AI1 Max ... AI3 Max): REAL (\leq 11 V or 22 mA) Minimum value of scaled output signal (AI1 Min scale ... AI3 Min scale): REAL Maximum value of scaled output signal (AI1 Max scale ... AI3 Max scale): REAL</p>																											
<p>Outputs</p>	<p>Analogue input mode (voltage or current) (AI1 mode ... AI3 mode): Boolean Value of analogue input (AI1 ... AI3): REAL Scaled value of analogue input (AI1 scaled ... AI3 scaled): REAL Error output (Error): DINT (0 = No error; 1 = Application program memory full)</p>																											

FIO_11_AO_slot1
(10090)

<p>Illustration</p>	
----------------------------	--

<p>Execution time</p>	<p>4.9 μs</p>
<p>Operation</p>	<p>The block controls the analogue output (AO1) of a FIO-11 Analog I/O Extension mounted on slot 1 of the drive control unit.</p> <p>The block converts the input signal (AO scaled) to a 0...20 mA signal (AO) that drives the analogue output; the input range AO Min Scale ... AO Max Scale corresponds to the current signal range of AO Min ... AO Max.</p> <p>AO Min Scale must be smaller than AO Max Scale; AO Max can be greater or smaller than AO Min.</p> <p>AO Min < AO Max</p>  <p>AO Min > AO Max</p> 
<p>Inputs</p>	<p>Minimum current signal (AO Min): REAL (0...20 mA) Maximum current signal (AO Max): REAL (0...20 mA) Minimum input signal (AO Min Scale): REAL Maximum input signal (AO Max Scale): REAL Input signal (AO scaled): REAL</p>
<p>Outputs</p>	<p>Analogue output current value (AO): REAL Error output (Error): DINT (0 = No error; 1 = Application program memory full)</p>

FIO_11_AO_slot2	
(10091)	
Illustration	<p>The diagram shows a block named 'FIO_11_AO_slot2' with a 'TLA1 1 msec (1)' parameter. It has five input lines on the left: 'AO Min', 'AO Max', 'AO Min Scale', 'AO Max Scale', and 'AO scaled'. It has two output lines on the right: 'AO(5:4)' and 'Error(5:4)'.</p>
Execution time	4.9 μ s
Operation	<p>The block controls the analogue output (AO1) of a FIO-11 Analog I/O Extension mounted on slot 2 of the drive control unit.</p> <p>The block converts the input signal (AO scaled) to a 0...20 mA signal (AO) that drives the analogue output; the input range AO Min Scale ... AO Max Scale corresponds to the current signal range of AO Min ... AO Max.</p> <p>AO Min Scale must be smaller than AO Max Scale; AO Max can be greater or smaller than AO Min.</p> <p>AO Min < AO Max</p> <p>AO Min > AO Max</p>

Inputs	Minimum current signal (AO Min): REAL (0...20 mA) Maximum current signal (AO Max): REAL (0...20 mA) Minimum input signal (AO Min Scale): REAL Maximum input signal (AO Max Scale): REAL Input signal (AO scaled): REAL
Outputs	Analogue output current value (AO): REAL Error output (Error): DINT (0 = No error; 1 = Application program memory full)

FIO_11_DIO_slot1 (10086)											
Illustration											
Execution time	6.0 μs										
Operation	<p>The block controls the two digital inputs/outputs (DIO1, DIO2) of a FIO-11 Digital I/O Extension mounted on slot 1 of the drive control unit.</p> <p>The state of a DIOx conf input of the block determines whether the corresponding DIO on the FIO-11 is an input or an output (0 = input, 1 = output). If the DIO is an output, the DOx input of the block defines its state.</p> <p>The DIx outputs show the state of the DIOs.</p> <p>The DIx filt gain inputs determine a filtering time for each input as follows:</p> <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>DIx filt gain</th> <th>Filtering time</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>7.5 μs</td> </tr> <tr> <td>1</td> <td>195 μs</td> </tr> <tr> <td>2</td> <td>780 μs</td> </tr> <tr> <td>3</td> <td>4.680 ms</td> </tr> </tbody> </table>	DIx filt gain	Filtering time	0	7.5 μs	1	195 μs	2	780 μs	3	4.680 ms
DIx filt gain	Filtering time										
0	7.5 μs										
1	195 μs										
2	780 μs										
3	4.680 ms										
Inputs	Digital input/output mode selection (DIO1 conf, DIO2 conf): Boolean Digital output state selection (DO1, DO2): Boolean Digital input filter gain selection (DI1 filt gain, DI2 filt gain): INT										
Outputs	Digital input/output state (DI1, DI2): Boolean Error output (Error): DINT (0 = No error; 1 = Application program memory full)										

FIO_11_DIO_slot2 (10087)											
Illustration											
Execution time	6.0 μ s										
Operation	<p>The block controls the two digital inputs/outputs (DIO1, DIO2) of a FIO-11 Digital I/O Extension mounted on slot 2 of the drive control unit.</p> <p>The state of a DIOx conf input of the block determines whether the corresponding DIO on the FIO-11 is an input or an output (0 = input, 1 = output). If the DIO is an output, the DOx input of the block defines its state.</p> <p>The Dlx outputs show the state of the DIOs.</p> <p>The Dlx filt gain inputs determine a filtering time for each input as follows:</p> <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Dlx filt gain</th> <th>Filtering time</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>7.5 μs</td> </tr> <tr> <td>1</td> <td>195 μs</td> </tr> <tr> <td>2</td> <td>780 μs</td> </tr> <tr> <td>3</td> <td>4.680 ms</td> </tr> </tbody> </table>	Dlx filt gain	Filtering time	0	7.5 μ s	1	195 μ s	2	780 μ s	3	4.680 ms
Dlx filt gain	Filtering time										
0	7.5 μ s										
1	195 μ s										
2	780 μ s										
3	4.680 ms										
Inputs	Digital input/output mode selection (DIO1 conf, DIO2 conf): Boolean Digital output state selection (DO1, DO2): Boolean Digital input filter gain selection (D11 filt gain, D12 filt gain): INT										
Outputs	Digital input/output state (D11, D12): Boolean Error output (Error): DINT (0 = No error; 1 = Application program memory full)										

Feedback & algorithms

CRITSPEED									
(10068)									
Illustration	<p>The diagram shows the CRITSPEED block with the following inputs and outputs:</p> <ul style="list-style-type: none"> Inputs: TLAI 1 msec (1), CRITSPEEDSEL, CRITSPEED1LO, CRITSPEED1HI, CRITSPEED2LO, CRITSPEED2HI, CRITSPEED3LO, CRITSPEED3HI, MAX, MIN, REFINPUT. Outputs: REFOUTPUT (57), OUTSTATE (57), OUTACTIVE(57). 								
Execution time	4.50 μ s								
Operation	<p>A critical speeds function block is available for applications where it is necessary to avoid certain motor speeds or speed bands because of e.g. mechanical resonance problems. The user can define three critical speeds or speed bands.</p> <p>Example: An application has vibrations in the range of 540 to 690 rpm and 1380 to 1560 rpm. To make the drive made to jump over the vibration speed ranges:</p> <ul style="list-style-type: none"> - activate the critical speeds function (CRITSPEEDSEL = 1), - set the critical speed ranges as in the figure below. <div style="display: flex; align-items: center;"> <div style="flex: 1;"> </div> <div style="flex: 1; border: 1px solid black; padding: 5px;"> <table border="1"> <tr><td>1</td><td>CRITSPEED1LO = 540 rpm</td></tr> <tr><td>2</td><td>CRITSPEED1HI = 690 rpm</td></tr> <tr><td>3</td><td>CRITSPEED2LO = 1380 rpm</td></tr> <tr><td>4</td><td>CRITSPEED2HI = 1560 rpm</td></tr> </table> </div> </div> <p>Output OUTACTIVE is set to 1 when the output reference (REFOUTPUT) is different from the input reference (REFINPUT).</p> <p>The output is limited by the defined minimum and maximum limits (MIN and MAX).</p> <p>Output OUTSTATE indicates in which critical speed range the operation point is.</p>	1	CRITSPEED1LO = 540 rpm	2	CRITSPEED1HI = 690 rpm	3	CRITSPEED2LO = 1380 rpm	4	CRITSPEED2HI = 1560 rpm
1	CRITSPEED1LO = 540 rpm								
2	CRITSPEED1HI = 690 rpm								
3	CRITSPEED2LO = 1380 rpm								
4	CRITSPEED2HI = 1560 rpm								
Inputs	<p>Critical speed activation input (CRITSPEEDSEL): Boolean</p> <p>Minimum/maximum critical speed range input (CRITSPEEDNLO / CRITSPEEDNHI): REAL</p> <p>Maximum/minimum input (MAX/MIN): REAL</p> <p>Reference input (REFINPUT): REAL</p>								
Outputs	<p>Reference output (REFOUTPUT): REAL</p> <p>Output state (OUTSTATE): REAL</p> <p>Output active (OUTACTIVE): Boolean</p>								

CYCLET	
(10074)	
Illustration	
Execution time	0.00 μ s
Operation	Output (OUT) is the time level of the CYCLET function block.
Inputs	-
Outputs	Output (OUT): DINT. 1 = 1 μ s

DATA CONTAINER	
(10073)	
Illustration	
Execution time	0.00 μ s
Operation	<p>Output (OUT) is an array of data with values 1...99. The array can be used by the XTAB and YTAB tables in block <i>FUNG-1V</i>. The array is defined by selecting "Define Pin Array Data" on the output pin in DriveSPC. Each value in the array must be on a separate row. Data can also be read from an *.arr file.</p> <p>Example:</p>
Inputs	-
Outputs	The output data type and the number of coordinate pairs are selected by the user. Output (OUT): DINT, INT, REAL or REAL24

FUNG-1V													
(10072)													
Illustration	<p>The diagram shows a rectangular block labeled 'FUNG-1V'. Inside the block, '(DINT)' is written above '60', and 'TLA1 1 msec' is written above '(1)'. On the left side, there are five input lines labeled BAL, BALREF, X, XTAB, and YTAB. On the right side, there are three output lines labeled Y(60), BALREFO(60), and ERROR(60). The output Y(60) is connected to a 'Y' label inside the block. The output BALREFO(60) is connected to a 'BALREFO' label inside the block. The output ERROR(60) is connected to an 'ERROR' label inside the block.</p>												
Execution time	9.29 μs												
Operation	<p>The output (Y) at the value of the input (X) is calculated with linear interpolation from a piecewise linear function.</p> $Y = Y_k + (X - X_k)(Y_{k+1} - Y_k) / (X_{k+1} - X_k)$ <p>The piecewise linear function is defined by the X and Y vector tables (XTAB and YTAB). For each X-value in the XTAB table, there is a corresponding Y-value in the YTAB table. The values in XTAB and YTAB must be in ascending order (i.e. from low to high).</p> <p>XTAB and YTAB values are defined with the DriveSPC tool.</p> <div style="display: flex; align-items: center;"> <div style="margin-right: 20px;"> <p>Interpolated Y</p> <p>The graph shows a coordinate system with X and Y axes. A piecewise linear function is plotted, consisting of four segments connecting points (X1, Y1), (X2, Y2), (X3, Y3), and (X4, Y4). Dashed lines indicate the interpolation between these points. The label 'Interpolated Y' is placed to the left of the Y-axis.</p> </div> <table border="1" style="border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">X table (XTAB)</th> <th style="text-align: left;">Y table (YTAB)</th> </tr> </thead> <tbody> <tr> <td>X1</td> <td>Y1</td> </tr> <tr> <td>X2</td> <td>Y2</td> </tr> <tr> <td>X3</td> <td>Y3</td> </tr> <tr> <td>...</td> <td>...</td> </tr> <tr> <td>X9</td> <td>Y9</td> </tr> </tbody> </table> </div> <p>The balancing function (BAL) permits the output signal to track an external reference and gives a smooth return to the normal operation. If BAL is set to 1, output Y is set to the value of the balance reference input (BALREF). The X value which corresponds to this Y value is calculated with linear interpolation and it is indicated by the balance reference output (BALREFO).</p> <p>If the X input is outside the range defined by the XTAB table, the output Y is set to the highest or lowest value in the YTAB table.</p> <p>If BALREF is outside the range defined by the YTAB table when balancing is activated (BAL: 0 → 1), the output Y is set to the value of the BALREF input and the BALREFO output is set to the highest or lowest value in the XTAB table.</p> <p>The ERROR output is set to 1 when the number of the XTAB and YTAB inputs are different. When ERROR is 1, the FUNG-1V block will not function. XTAB and YTAB tables can be defined in the DATA CONTAINER block or the REG-G block.</p>	X table (XTAB)	Y table (YTAB)	X1	Y1	X2	Y2	X3	Y3	X9	Y9
X table (XTAB)	Y table (YTAB)												
X1	Y1												
X2	Y2												
X3	Y3												
...	...												
X9	Y9												
Inputs	<p>The input data type is selected by the user.</p> <p>Balance input (BAL): Boolean</p> <p>Balance reference input (BALREF): DINT, INT, REAL, REAL24.</p> <p>X value input (X): DINT, INT, REAL, REAL24</p> <p>X table input (XTAB): DINT, INT, REAL, REAL24</p> <p>Y table input (YTAB): DINT, INT, REAL, REAL24</p>												
Outputs	<p>Y value output (Y): DINT, INT, REAL, REAL24</p> <p>Balance reference output (BALREFO): DINT, INT, REAL, REAL24</p> <p>Error output (ERROR): Boolean</p>												

INT	
(10065)	
Illustration	
Execution time	4.73 μs
Operation	<p>The output (O) is the integrated value of the input (I):</p> $O(t) = K/TI \left(\int I(t) dt \right)$ <p>Where TI is the integration time constant and K is the integration gain.</p> <p>The step response for the integration is:</p> $O(t) = K \times I(t) \times t/TI$ <p>The transfer function for the integration is:</p> $G(s) = K \ 1/sTI$ <p>The output value is limited according to the defined minimum and maximum limits (OLL and OHL). If the value is below the minimum value, output O = LL is set to 1. If the value exceeds the maximum value, output O = HL is set to 1. The output (O) retains its value when the input signal I(t) = 0.</p> <p>The integration time constant is limited to value 2147483 ms. If the time constant is negative, zero time constant is used.</p> <p>If the ratio between the cycle time and the integration time constant $Ts/TI < 1$, Ts/TI is set to 1.</p> <p>The integrator is cleared when the reset input (RINT) is set to 1.</p> <p>If BAL is set to 1, output O is set to the value of the input BALREF. When BAL is set back to 0, normal integration operation continues.</p>
Inputs	<p>Input (I): REAL</p> <p>Gain input (K): REAL</p> <p>Integration time constant input (TI): DINT, 0...2147483 ms</p> <p>Integrator reset input (RINT): Boolean</p> <p>Balance input (BAL): Boolean</p> <p>Balance reference input (BALREF): REAL</p> <p>Output high limit input (OHL): REAL</p> <p>Output low limit input (OLL): REAL</p>
Outputs	<p>Output (O): REAL</p> <p>High limit output (O=HL): Boolean</p> <p>Low limit output (O=LL): Boolean</p>

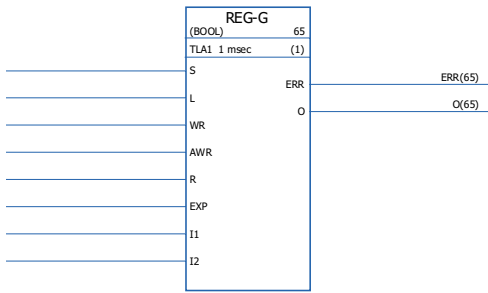
MOTPOT (10067)	
Illustration	
Execution time	2.92 μ s
Operation	<p>The motor potentiometer function controls the rate of change of the output from the minimum to the maximum value and vice versa.</p> <p>The function is enabled by setting the ENABLE input to 1. If the up input (UP) is 1, the output reference (OUTPUT) is increased to the maximum value (MAXVAL) with the defined ramp time (RAMPTIME). If the down input (DOWN) is 1, the output value is decreased to the minimum value (MINVAL) with the defined ramp time. If the up and down inputs are activated/deactivated simultaneously, the output value is not increased/decreased.</p> <p>If the RESET input is 1, the output will be reset to the value defined by the reset value input (RESETVAL) or to the value defined by the minimum input (MINVAL), whichever is higher.</p> <p>If the ENABLE input is 0, the output is zero.</p> <p>Digital inputs are normally used as up and down inputs.</p>
Inputs	Function enable input (ENABLE): Boolean Up input (UP): Boolean Down input (DOWN): Boolean Ramp time input (RAMPTIME): REAL (seconds) (i.e. the time required for the output to change from the minimum to the maximum value or from the maximum to the minimum value) Maximum reference input (MAXVAL): REAL Minimum reference input (MINVAL): REAL Reset value input (RESETVAL): REAL Reset input (RESET): Boolean
Outputs	Output (OUTPUT) REAL

PID (10075)							
Illustration							
Execution time	15.75 μs						
Operation	<p>The PID controller can be used for closed-loop control systems. The controller includes anti-windup correction and output limitation.</p> <p>The PID controller output (Out) before limitation is the sum of the proportional (U_P), integral (U_I) and derivative (U_D) terms:</p> $Out_{unlimited}(t) = U_P(t) + U_I(t) + U_D(t)$ $U_P(t) = P \times Dev(t)$ $U_I(t) = P/tI \times \left[\int Dev(\tau) d\tau + tC \times (Out(t) - Out_{unlimited}(t)) \right]$ $U_D(t) = P \times tD \times d(Dev(t))/dt$ <p>Integrator:</p> <p>The integral term can be cleared by setting I_reset to 1. Note that the anti-windup correction is simultaneously disabled. When I_reset is 1, the controller acts as a PD controller.</p> <p>If integration time constant tI is 0, the integral term will not be updated.</p> <p>Smooth return to normal operation is guaranteed after errors or abrupt input value changes. This is achieved by adjusting the integral term so that the output will retain its previous value during these situations.</p> <p>Limitation:</p> <p>The output is limited by the defined minimum and maximum values, OLL and OHL:</p> <p>If the actual value of the output reaches the specified minimum limit, output O=LL is set to 1.</p> <p>If the actual value of the output reaches the specified maximum limit, output O=HL is set to 1.</p> <p>Smooth return to normal operation after limitation is requested if and only if the anti-windup correction is not used, i.e. when tI = 0 or tC = 0.</p> <p>Error codes:</p> <p>Error codes are indicated by the error output (ERROR) as follows</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">Error code</th> <th style="text-align: left;">Description</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1</td> <td>The minimum limit (OLL) exceeds the maximum limit (OHL).</td> </tr> <tr> <td style="text-align: center;">2</td> <td>Overflow with U_p, U_i, or U_d calculation</td> </tr> </tbody> </table>	Error code	Description	1	The minimum limit (OLL) exceeds the maximum limit (OHL).	2	Overflow with U_p , U_i , or U_d calculation
Error code	Description						
1	The minimum limit (OLL) exceeds the maximum limit (OHL).						
2	Overflow with U_p , U_i , or U_d calculation						

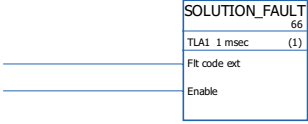
	<p>Balancing:</p> <p>The balancing function (BAL) permits the output signal to track an external reference and gives a smooth return to the normal operation. If BAL is set to 1, the output (Out) is set to the value of the balance reference input (BAL_ref). Balance reference is limited by the defined minimum and maximum limits (OLL and OHL).</p> <p>Anti-windup:</p> <p>Anti-windup correction time constant is defined by input tC, which defines the time after which the difference between the unlimited and limited outputs is subtracted from the I-term during limitation. If tC = 0 or tI = 0, anti-windup correction is disabled.</p>
Inputs	<p>Actual input (IN_act): REAL</p> <p>Reference input (IN_ref): REAL</p> <p>Proportional gain input (P): REAL</p> <p>Integration time constant input (tI): REAL. 1 = 1 ms</p> <p>Derivation time constant input (tD): REAL. 1 = 1 ms</p> <p>Antiwind-up correction time constant input (tC): IQ6. 1 = 1 ms</p> <p>Integrator reset input (I_reset): Boolean</p> <p>Balance input (BAL): Boolean</p> <p>Balance reference input (BAL_ref): REAL</p> <p>Output high limit input (OHL): REAL</p> <p>Output low limit input (OLL): REAL</p>
Outputs	<p>Output (Out): REAL</p> <p>Deviation output (Dev): REAL (= actual -reference = IN_act - IN_ref)</p> <p>High limit output (O=HL): Boolean</p> <p>Low limit output (O=LL): Boolean</p> <p>Error code output (ERROR): INT32</p>

<p>RAMP</p> <p>(10066)</p>	
Illustration	<pre> graph LR subgraph RAMP [RAMP 64] direction TB TLA1[TLA1 1 msec (1)] IN[IN] STEP+[STEP+] STEP-[STEP-] SLOPE+[SLOPE+] SLOPE-[SLOPE-] BAL[BAL] BALREF[BALREF] OHL[OHL] OLL[OLL] end IN --- O64[O(64)] STEP+ --- OHL64[O=HL(64)] STEP- --- OLL64[O=LL(64)] SLOPE+ --- OHL64 SLOPE- --- OLL64 BAL --- OHL64 BALREF --- OHL64 OHL --- OHL64 OLL --- OLL64 </pre>
Execution time	4.23 μs

<p>Operation</p>	<p>Limits the rate of the change of the signal.</p> <p>The input signal (IN) is connected directly to the output (O) if the input signal does not exceed the defined step change limits (STEP+ and STEP-). If the input signal change exceeds these limits, the output signal change is limited by the maximum step change (STEP+/STEP- depending on the direction of rotation). After this, the output signal is accelerated/decelerated by the defined ramp value (SLOPE+/SLOPE-) per second until the input and output signal values are equal.</p> <p>The output is limited by the defined minimum and maximum values (OLL and OHL). If the actual value of the output falls below the specified minimum limit (OLL), output O=LL is set to 1. If the actual value of the output exceeds the specified maximum limit (OHL), output O=HL is set to 1.</p> <p>If the balancing input (BAL) is set to 1, the output (O) is set to the value of the balance reference input (BAL_ref). Balancing reference is also limited by the minimum and maximum values (OLL and OHL).</p>
<p>Inputs</p>	<p>Input (IN): REAL</p> <p>Maximum positive step change input (STEP+): REAL</p> <p>Maximum negative step change input (STEP-): REAL</p> <p>Ramp-up value per second input (SLOPE+): REAL</p> <p>Ramp-down value per second input (SLOPE-): REAL</p> <p>Balance input (BAL): Boolean</p> <p>Balance reference input (BALREF): REAL</p> <p>Output high limit input (OHL): REAL</p> <p>Output low limit input (OLL): REAL</p>
<p>Outputs</p>	<p>Output (O): REAL</p> <p>High limit output (O=HL): Boolean</p> <p>Low limit output (O=LL): Boolean</p>

<p>REG-G (10102)</p>	
<p>Illustration</p>	
<p>Execution time</p>	<p>-</p>

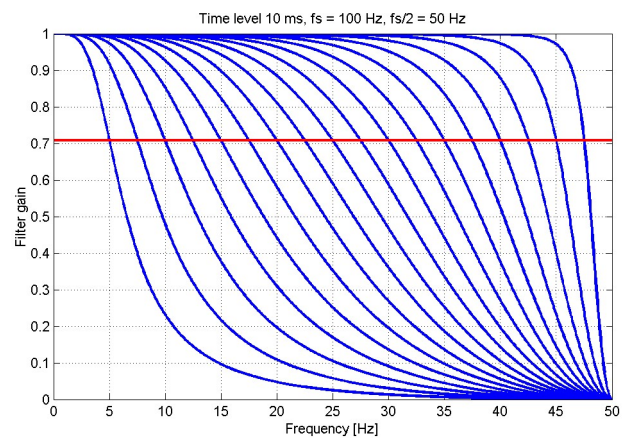
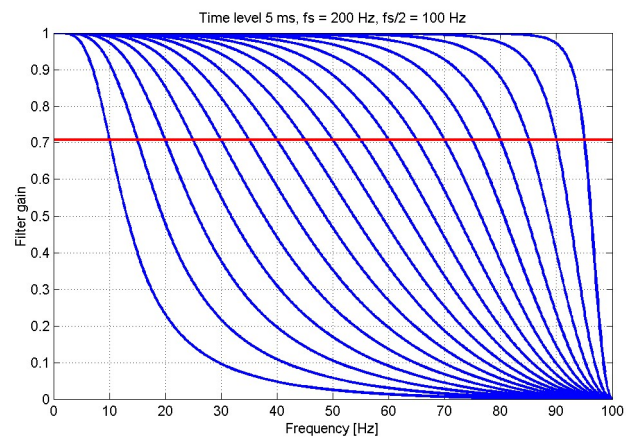
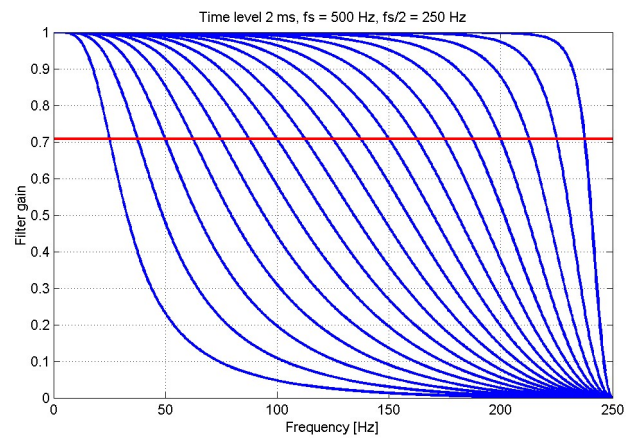
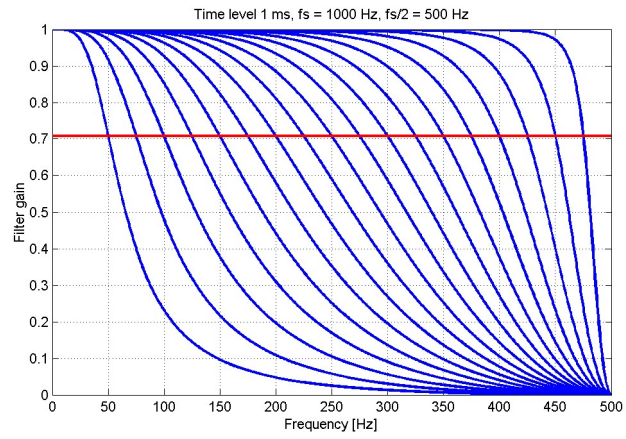
<p>Operation</p>	<p>Combines the array (group of variables) (if any) on the EXP input with the values of the I1...I32 pins to produce an output array. The data type of the arrays can be INT, DINT, REAL16, REAL24 or Boolean. The output array consists of the data from the EXP input and the values of the I1...In (in this order).</p> <p>When input S is 1, data is continuously assembled into the output array. The element acts as a latch when input S is 0; the latest data assembled then remains at the output.</p> <p>If S is 0 and L changes state from 0 to 1, the array from the EXP input and the values of the I1...In inputs are copied to output O during this program cycle. If S or R is 1, L has no effect.</p> <p>WR and AWR are used to change individual cells of the output array. AWR indicates the input whose value is moved to the output array. If AWR is 0, only the array from input EXP is moved to the output. If AWR is not 0, the corresponding I input is moved to the output. This is performed when WR goes from 0 to 1.</p> <p>When input R is 1, the output array is cleared and all further data entry is prevented. R overrides both S and L. If WR is 1, the address at AWR is checked and if it is illegal (negative or greater than the number of inputs), the error output (ERR) is set to 2. Otherwise ERR is 0.</p> <p>Whenever an error is detected, ERR is set within one cycle. No place in the register is affected when an error occurs.</p> <p>Example:</p> <p>In the diagram, the DATA CONTAINER block includes an array with values [1,2,3,4]. At start, the output array is [0,0,0,0,0,0,0,0]. When WR changes to 1 and returns to 0, the AWR value of 0 means that only EXP is moved into the output array, which now reads [1,2,3,4,0,0,0,0]. After this, AWR is changed to 3, meaning that inputs EXP and I3 are moved to the output. After a WR switch, the output array is [1,2,3,4,0,7,0].</p>
<p>Inputs</p>	<p>Set (S): Boolean, INT, DINT, REAL, REAL24 Load (L): Boolean, INT, DINT, REAL, REAL24 Write (WR): Boolean, INT, DINT, REAL, REAL24 Write address (AWR): INT Reset (R): Boolean Expander (EXP): IArray Data input (I1...I32): Boolean, INT, DINT, REAL, REAL24</p>
<p>Outputs</p>	<p>Error (ERR): INT Array data output (O): OC1</p>

SOLUTION_FAULT	
(10097)	
Illustration	 <p>The diagram shows a rectangular block labeled 'SOLUTION_FAULT' with the number '66' in the top right corner. Three horizontal lines represent inputs: the top line is labeled 'TLA1 1 msec (1)', the middle line is labeled 'Flt code ext', and the bottom line is labeled 'Enable'.</p>
Execution time	-
Operation	When the block is enabled (by setting the Enable input to 1), a fault (F-0317 SOLUTION FAULT) is generated by the drive. The value of the Flt code ext input is recorded by the fault logger.
Inputs	Fault code extension (Flt code ext): DINT Generate fault (Enable): Boolean
Outputs	-

Filters

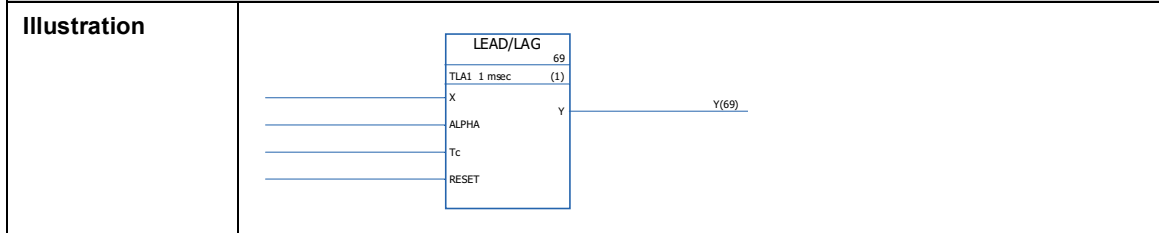
FILT1 (10069)	
Illustration	
Execution time	7.59 μs
Operation	<p>The output (O) is the filtered value of the input (I) value and the previous output value (O_{prev}). The FILT1 block acts as 1st order low pass filter.</p> <p>Note: Filter time constant (T1) must be selected so that T1/Ts < 32767. If the ratio exceeds 32767, it is considered as 32767. Ts is the cycle time of the program in ms. If T1 < Ts, the output value is the input value.</p> <p>The step response for a single pole low pass filter is:</p> $O(t) = I(t) \times (1 - e^{-t/T1})$ <p>The transfer function for a single pole low pass filter is:</p> $G(s) = 1 / (1 + sT1)$
Inputs	<p>Input (I): REAL</p> <p>Filter time constant input (T1): DINT, 1 = 1 ms</p>
Outputs	Output (O): REAL

FILT2 (10070)	
Illustration	
Execution time	6.30 μs
Operation	<p>The output (Y) is the filtered value of the input (X). The FILT2 block acts as a 2nd order low pass filter.</p> <p>When the RESET input value is set to 1, the input is connected to the output without filtering.</p> <p>Notes:</p> <ul style="list-style-type: none"> • The -3 dB cutoff frequency (FRQ) is limited to its maximum value (16383 Hz). • The frequency of the input signal must be less than half of sampling frequency (fs) – any higher frequencies are aliased to the allowable range. The sampling frequency is defined by the time level of the block; for example, 1 ms corresponds to a sampling frequency of 1000 Hz. <p>The following diagrams show the frequency responses for 1, 2, 5 and 10 ms time levels. The -3 dB cutoff level is represented as the horizontal line at 0.7 gain.</p>



Inputs	Input (X): REAL -3 dB cutoff frequency input (FRQ): DINT (0...16383 Hz) Reset input (RESET): Boolean
Outputs	Output (Y): REAL

LEAD/LAG
(10071)



Execution time	5.55 μs
Operation	<p>The output (Y) is the filtered value of the input (X). When ALPHA > 1, the function block acts as a lead filter. When ALPHA < 1, the function block acts as a lag filter. When ALPHA = 1, no filtering occurs.</p> <p>The transfer function for a lead/lag filter is: $(1 + \text{ALPHA}T_c s) / (1 + T_c s)$</p> <p>When RESET input is 1, the input value (X) is connected to the output (Y). If ALPHA or Tc < 0, the negative input value is set to zero before filtering.</p>
Inputs	Input (X): REAL Lead/Lag filter type input (ALPHA): REAL Time constant input (Tc): REAL Reset input (RESET): Boolean
Outputs	Output (Y): REAL

Parameters

GetBitPtr (10099)	
Illustration	
Execution time	-
Operation	Reads the status of one bit within a parameter value cyclically. The Bit ptr input specifies the parameter group, index and bit to be read. The output (Out) provides the value of the bit.
Inputs	Parameter group, index and bit (Bit ptr): DINT
Outputs	Bit status (Out): DINT

GetValPtr (10098)	
Illustration	
Execution time	-
Operation	Reads the value of a parameter cyclically. The Par ptr input specifies the parameter group and index to be read. The output (Out) provides the value of the parameter.
Inputs	Parameter group and index (Par ptr): DINT
Outputs	Parameter value (Out): DINT

PARRD (10082)							
Illustration							
Execution time	6.00 μ s						
Operation	<p>Reads the scaled value of a parameter (specified by the Group and Index inputs). If the parameter is a pointer parameter, the Output pin provides the number of the source parameter instead of its value.</p> <p>Error codes are indicated by the error output (Error) as follows:</p> <table border="1"> <thead> <tr> <th>Error code</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>No error</td> </tr> <tr> <td><> 0</td> <td>Error</td> </tr> </tbody> </table> <p>See also blocks PARRDINTR and PARRDPTR.</p>	Error code	Description	0	No error	<> 0	Error
Error code	Description						
0	No error						
<> 0	Error						

Inputs	Parameter group input (Group): DINT Parameter index input (Index): DINT
Outputs	Output (Output): DINT Error output (Error): DINT

PARRDINTR (10101)							
Illustration	<p>The diagram shows a rectangular block labeled 'PARRDINTR (BOOL) 73'. It has two input lines on the left labeled 'Group' and 'Index'. It has two output lines on the right labeled 'Output(73)' and 'Error(73)'. Above the block, it specifies 'TLAI 1 msec (1)'.</p>						
Execution time	-						
Operation	<p>Reads the internal (non-scaled) value of a parameter (specified by the Group and Index inputs). The value is provided by the Output pin.</p> <p>Error codes are indicated by the error output (Error) as follows:</p> <table border="1"> <thead> <tr> <th>Error code</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>No error or busy</td> </tr> <tr> <td><> 0</td> <td>Error</td> </tr> </tbody> </table> <p>Note: Using this block may cause incompatibility issues when upgrading the application to another firmware version.</p>	Error code	Description	0	No error or busy	<> 0	Error
Error code	Description						
0	No error or busy						
<> 0	Error						
Inputs	Parameter group (Group): DINT Parameter index (Index): DINT						
Outputs	Output (Output): Boolean, INT, DINT, REAL, REAL24 Error output (Error): DINT						

PARRDPTR (10100)							
Illustration	<p>The diagram shows a rectangular block labeled 'PARRDPTR (BOOL) 74'. It has two input lines on the left labeled 'Group' and 'Index'. It has two output lines on the right labeled 'Output(74)' and 'Error(74)'. Above the block, it specifies 'TLAI 1 msec (1)'.</p>						
Execution time	-						
Operation	<p>Reads the internal (non-scaled) value of the source of a pointer parameter. The pointer parameter is specified using the Group and Index inputs.</p> <p>The value of the source selected by the pointer parameter is provided by the Output pin.</p> <p>Error codes are indicated by the error output (Error) as follows:</p> <table border="1"> <thead> <tr> <th>Error code</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>No error or busy</td> </tr> <tr> <td><> 0</td> <td>Error</td> </tr> </tbody> </table>	Error code	Description	0	No error or busy	<> 0	Error
Error code	Description						
0	No error or busy						
<> 0	Error						

Inputs	Parameter group (Group): DINT Parameter index (Index): DINT
Outputs	Output (Output): Boolean, INT, DINT, REAL, REAL24 Error output (Error): DINT

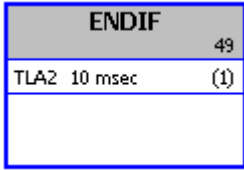
PARWR (10080)							
Illustration							
Execution time	14.50 μs						
Operation	<p>The input value (IN) is written to the defined parameter (Group and Index). The new parameter value is stored to the flash memory if the store input (Store) is 1. Note: Cyclic parameter value storing can damage the memory unit. Parameter values should be stored only when necessary. Error codes are indicated by the error output (Error) as follows:</p> <table border="1"> <thead> <tr> <th>Error code</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>No error</td> </tr> <tr> <td><> 0</td> <td>Error</td> </tr> </tbody> </table>	Error code	Description	0	No error	<> 0	Error
Error code	Description						
0	No error						
<> 0	Error						
Inputs	Input (IN): DINT Parameter group input (Group): DINT Parameter index input (Index): DINT Store input (Store): Boolean						
Outputs	Error output (Error): DINT						

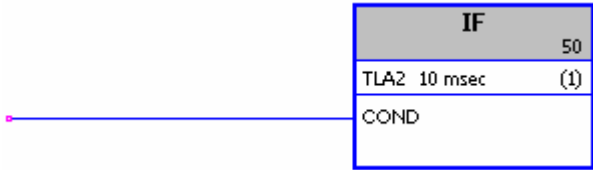
Program structure

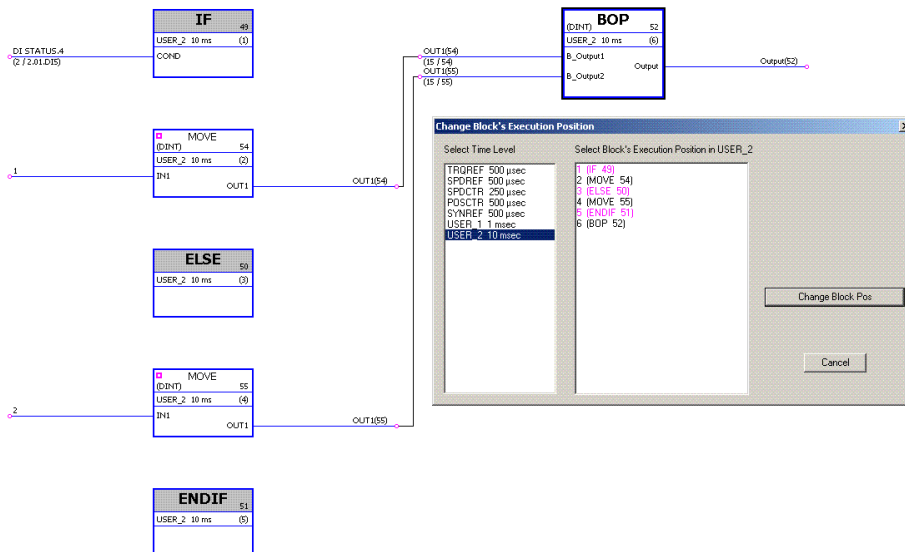
BOP	
(10105)	
Illustration	
Execution time	-
Operation	<p>The BOP (Bundle OutPut) block collects the outputs of several different sources. The sources are connected to the B_Output pins. The B_Output pin that changed last is relayed to the Output pin.</p> <p>The block is intended for use with conditional IF-ENDIF structures. See the example under the <i>IF</i> block.</p>
Inputs	Values from different conditional branches (B_Output1...B_OutputN): INT, DINT, Boolean, REAL, REAL24
Outputs	Output from currently active branch of a IF-ELSEIF structure or latest updated input value (Output): INT, DINT, Boolean, REAL, REAL24

ELSE	
Illustration	
Execution time	-
Operation	See the description of the <i>IF</i> block.
Inputs	-
Outputs	-

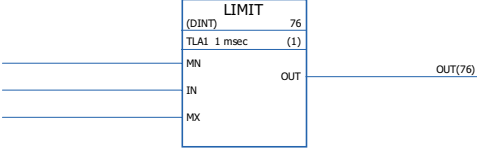
ELSEIF	
Illustration	
Execution time	-
Operation	See the description of the <i>IF</i> block.
Inputs	Input (COND): Boolean
Outputs	-

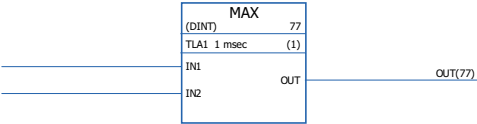
ENDIF	
Illustration	 <p>The diagram shows a rectangular block with a grey header containing the text 'ENDIF' and the number '49'. Below the header, the text 'TLA2 10 msec' is followed by '(1)' in parentheses. The block is outlined in blue.</p>
Execution time	-
Operation	See the description of the <i>IF</i> block.
Inputs	-
Outputs	-

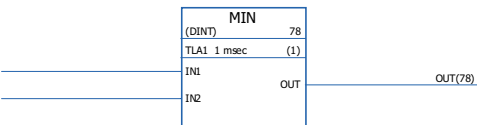
IF	
(10103)	
Illustration	 <p>The diagram shows a rectangular block with a grey header containing the text 'IF' and the number '50'. Below the header, the text 'TLA2 10 msec' is followed by '(1)' in parentheses. Underneath, the text 'COND' is present. A blue line with a small square at its end extends from the left side of the block to the 'COND' text. The block is outlined in blue.</p>
Execution time	-

<p>Operation</p>	<p>The IF, ELSE, ELSEIF and ENDIF blocks define, by Boolean logic, which parts of the application program are executed.</p> <p>If the condition input (COND) is true, the blocks between the IF block and the next ELSEIF, ELSE or ENDIF block (in execution order) are run. If the condition input (COND) is false, the blocks between the IF block and the next ELSEIF, ELSE or ENDIF block are skipped.</p> <p>The outputs of the “branches” are collected and selected by using the <i>BOP</i> block.</p> <p>Example:</p> <p>Bit 4 of 02.01 DI status (digital input DI5) controls the branching of the application program. If the input is 0, the blocks between the IF and ELSE blocks are skipped but the blocks between ELSE and ENDIF are run. If the input is 1, the blocks between IF and ELSE are run. The program execution then jumps to the block that follows ENDIF, which is a BOP. The BOP block outputs the value from the branch that was executed. If the digital input is 0, the BOP block output is 2; if the digital input is 1, the BOP block output is 1.</p> 
<p>Inputs</p>	<p>Input (COND): Boolean</p>
<p>Outputs</p>	<p>-</p>

Selection

LIMIT (10052)	
Illustration	
Execution time	0.53 μ s
Operation	The output (OUT) is the limited input (IN) value. Input is limited according to the minimum (MN) and maximum (MX) values.
Inputs	The input data type is selected by the user. Minimum input limit (MN): INT, DINT, REAL, REAL24 Input (IN): INT, DINT, REAL, REAL24 Maximum input limit (MX): INT, DINT, REAL, REAL24
Outputs	Output (OUT): INT, DINT, REAL, REAL24

MAX (10053)	
Illustration	
Execution time	0.81 μ s (when two inputs are used) + 0.53 μ s (for every additional input). When all inputs are used, the execution time is 16.73 μ s.
Operation	The output (OUT) is the highest input value (IN).
Inputs	The input data type and the number of inputs (2...32) are selected by the user. Input (IN1...IN32): INT, DINT, REAL, REAL24
Outputs	Output (OUT): INT, DINT, REAL, REAL24

MIN (10054)	
Illustration	
Execution time	0.81 μ s (when two inputs are used) + 0.52 μ s (for every additional input). When all inputs are used, the execution time is 16.50 μ s.
Operation	The output (OUT) is the lowest input value (IN).
Inputs	The input data type and the number of inputs (2...32) are selected by the user. Input (IN1...IN32): INT, DINT, REAL, REAL24

Outputs	Output (OUT): INT, DINT, REAL, REAL24
----------------	---------------------------------------

MUX (10055)	
Illustration	
Execution time	0.70 μs
Operation	The value of an input (IN) selected by the address input (K) is stored to the output (OUT). If the address input is 0, negative or exceeds the number of the inputs, the output is 0.
Inputs	The input data type and number of inputs (2...32) are selected by the user. Address input (K): DINT Input (IN1...IN32): INT, DINT, REAL, REAL24
Outputs	Output (OUT): INT, DINT, REAL, REAL24

SEL (10056)	
Illustration	
Execution time	1.53 μs
Operation	The output (OUT) is the value of the input (IN) selected by the selection input (G). If G = 0: OUT = IN A. If G = 1: OUT = IN B.
Inputs	The input data type is selected by the user. Selection input (G): Boolean Input (IN A, IN B): Boolean, INT, DINT, REAL, REAL24
Outputs	Output (OUT): Boolean, INT, DINT, REAL, REAL24

Switch & Demux

DEMUX-I (10061)	
Illustration	
Execution time	1.38 μ s (when two outputs are used) + 0.30 μ s (for every additional output). When all outputs are used, the execution time is 10.38 μ s.
Operation	Input (I) value is stored to the output (OA1...OA32) selected by the address input (A). All other outputs are 0. If the address input is 0, negative or exceeds the number of the outputs, all outputs are 0.
Inputs	The input data type is selected by the user. Address input (A): DINT Input (I): INT, DINT, Boolean, REAL, REAL24
Outputs	The number of the output channels (1...32) is selected by the user. Output (OA1...OA32): INT, DINT, REAL, REAL24, Boolean

DEMUX-MI (10062)																																																																
Illustration																																																																
Execution time	0.99 μ s (when two outputs are used) + 0.25 μ s (for every additional output). When all outputs are used, the execution time is 8.4 μ s.																																																															
Operation	<p>The input (I) value is stored to the output (OA1...OA32) selected by the address input (A) if the load input (L) or the set input (S) is 1. When the load input is set to 1, the input (I) value is stored to the output only once. When the set input is set to 1, the input (I) value is stored to the output every time the block is executed. The set input overrides the load input.</p> <p>If the reset input (R) is 1, all connected outputs are 0.</p> <p>If the address input is 0, negative or exceeds the number of the outputs, all outputs are 0.</p> <p>Example:</p> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th>S</th> <th>L</th> <th>R</th> <th>A</th> <th>I</th> <th>OA1</th> <th>OA2</th> <th>OA3</th> <th>OA4</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>0</td> <td>0</td> <td>2</td> <td>150</td> <td>0</td> <td>150</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>0</td> <td>0</td> <td>2</td> <td>120</td> <td>0</td> <td>150</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>3</td> <td>100</td> <td>0</td> <td>150</td> <td>100</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>1</td> <td>200</td> <td>200</td> <td>150</td> <td>100</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>4</td> <td>250</td> <td>200</td> <td>150</td> <td>100</td> <td>250</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>2</td> <td>300</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> </tbody> </table>	S	L	R	A	I	OA1	OA2	OA3	OA4	1	0	0	2	150	0	150	0	0	0	0	0	2	120	0	150	0	0	0	1	0	3	100	0	150	100	0	1	0	0	1	200	200	150	100	0	1	1	0	4	250	200	150	100	250	1	1	1	2	300	0	0	0	0
S	L	R	A	I	OA1	OA2	OA3	OA4																																																								
1	0	0	2	150	0	150	0	0																																																								
0	0	0	2	120	0	150	0	0																																																								
0	1	0	3	100	0	150	100	0																																																								
1	0	0	1	200	200	150	100	0																																																								
1	1	0	4	250	200	150	100	250																																																								
1	1	1	2	300	0	0	0	0																																																								

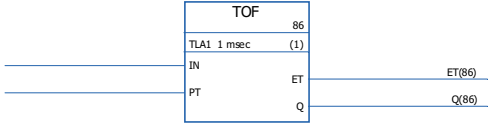
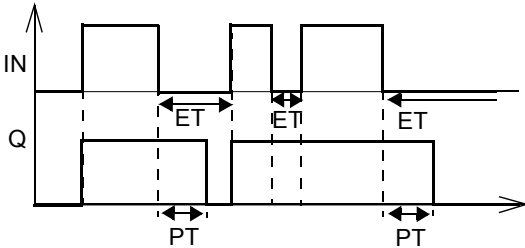
Inputs	The input data type is selected by the user. Address input (A): DINT Reset input (R): Boolean Load input (L): Boolean Set input (S): Boolean Input (I): DINT, INT, REAL, REAL24, Boolean
Outputs	The number of the output channels (1...32) is selected by the user. Output (OA1...OA32): DINT, INT, REAL, REAL24, Boolean

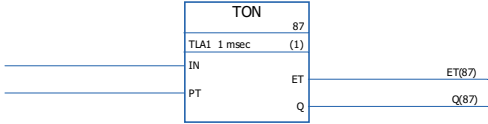
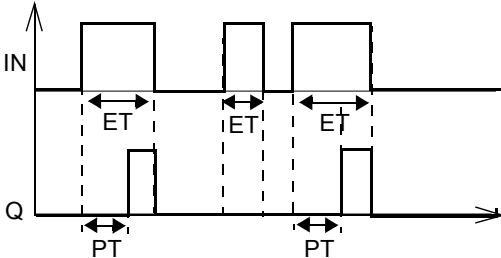
SWITCH (10063)	
Illustration	
Execution time	0.68 μ s (when two inputs are used) + 0.50 μ s (for every additional input). When all inputs are used, the execution time is 15.80 μ s.
Operation	The output (OUT) is equal to the corresponding input (IN) if the activate input (ACT) is 1. Otherwise the output is 0.
Inputs	The input data type and the number of inputs (1...32) are selected by the user. Activate input (ACT): Boolean Input (IN1...IN32): INT, DINT, REAL, REAL24, Boolean
Outputs	Output (OUT1...OUT32): INT, DINT, REAL, REAL24, Boolean

SWITCHC (10064)	
Illustration	
Execution time	1.53 μ s (when two inputs are used) + 0.73 μ s (for every additional input). When all inputs are used, the execution time is 23.31 μ s.
Operation	The output (OUT) is equal to the corresponding channel A input (CH A1...32) if the activate input (ACT) is 0. The output is equal to the corresponding channel B input (CH B1...32) if the activate input (ACT) is 1.
Inputs	The input data type and the number of inputs (1...32) are selected by the user. Activate input (ACT): Boolean Input (CH A1...CH A32, CH B1...CH B32): INT, DINT, REAL, REAL24, Boolean
Outputs	Output (OUT1...OUT32): INT, DINT, REAL, REAL24, Boolean

Timers

MONO (10057)	
Illustration	
Execution time	1.46 μ s
Operation	<p>The output (O) is set to 1 and the timer is started, if the input (I) is set to 1. The output is reset to 0 when the time defined by the time pulse input (TP) has elapsed. Elapsed time (TE) count starts when the output is set to 1 and stops when the output is set to 0.</p> <p>If RTG is 0, a new input pulse during the time defined by TP has no effect on the function. The function can be restarted only after the time defined by TP has elapsed.</p> <p>If RTG is 1, a new input pulse during the time defined by TP restarts the timer and sets the elapsed time (TE) to 0.</p> <p>Example 1: MONO is not re-triggable, i.e. RTG = 0.</p> <p>Example 2: MONO is re-triggable, i.e. RTG = 1.</p>
Inputs	Re-trigger input (RTG): Boolean Time pulse input (TP): DINT (1 = μ s) Input (I): Boolean
Outputs	Output (O): Boolean Time elapsed output (TE): DINT (1 = 1 μ s)

TOF (10058)	
Illustration	
Execution time	1.10 μ s
Operation	<p>The output (Q) is set to 1, when the input (IN) is set to 1. The output is reset to zero when the input has been 0 for a time defined by the pulse time input (PT).</p> <p>Elapsed time count (ET) starts when the input is set to 0 and stops when the input is set to 1.</p> <p>Example:</p> 
Inputs	Input (IN): Boolean Pulse time input (PT): DINT (1 = 1 μ s)
Outputs	Elapsed time output (ET): DINT (1 = 1 μ s) Output (Q): Boolean

TON (10059)	
Illustration	
Execution time	1.22 μ s
Operation	<p>The output (Q) is set to 1 when the input (IN) has been 1 for a time defined by the pulse time input (PT). The output is set to 0, when the input is set to 0.</p> <p>Elapsed time count (ET) starts when the input is set to 1 and stops when the input is set to 0.</p> <p>Example:</p> 

Inputs	Input (IN): Boolean Pulse time input (PT): DINT (1 = 1 μ s)
Outputs	Elapsed time output (ET): DINT (1 = 1 μ s) Output (Q): Boolean

TP (10060)	
Illustration	
Execution time	1.46 μ s
Operation	<p>The output (Q) is set to 1 when the input (IN) is set to 1. The output is set to 0, when it has been 1 for a time defined by the pulse time input (PT).</p> <p>Elapsed time count (ET) starts when the input is set to 1 and stops when the input is set to 0.</p>
Inputs	Pulse time input (PT): DINT (1 = 1 μ s) Input (IN): Boolean
Outputs	Output (Q): Boolean Elapsed time output (ET): DINT (1 = 1 μ s)





Examples of using standard function blocks

What this chapter contains

This chapter contains examples of using standard function blocks for

- start/stop
- relay output and digital input/output control
- drive-to-drive communication.

Start/stop using analog input

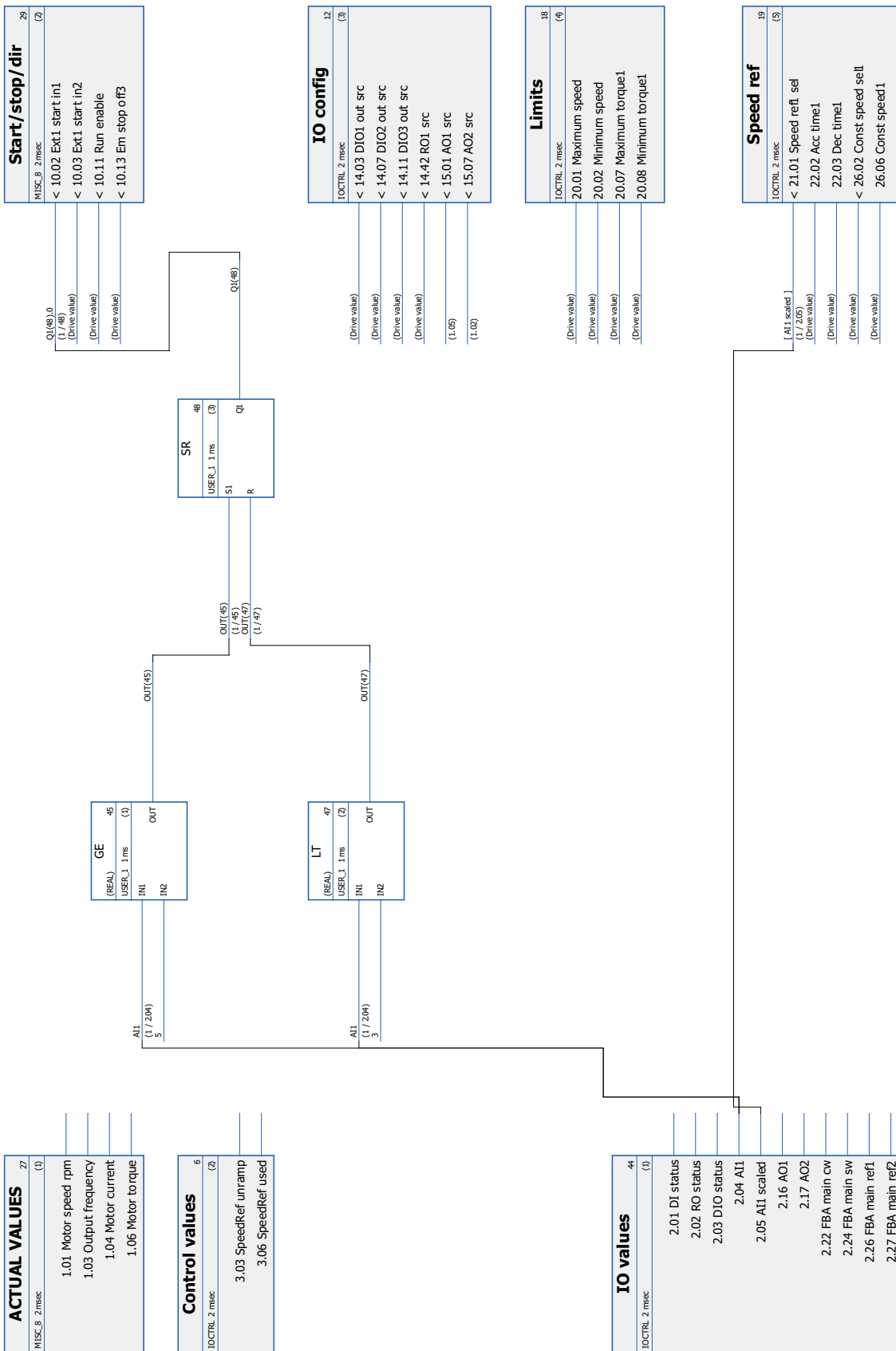
This example presents an application program, where

- the speed reference is given via analog input AI1
- the drive starts when AI1 is higher than 5 mA
- the drive stops when AI1 is lower than 3 mA.

Additional information

- Actual signal *02.04 AI1* displays AI1 as measured.
 - The program is executed at the dedicated time level of 1 ms.
-

92 Examples of using standard function blocks



Relay output and digital input/output control

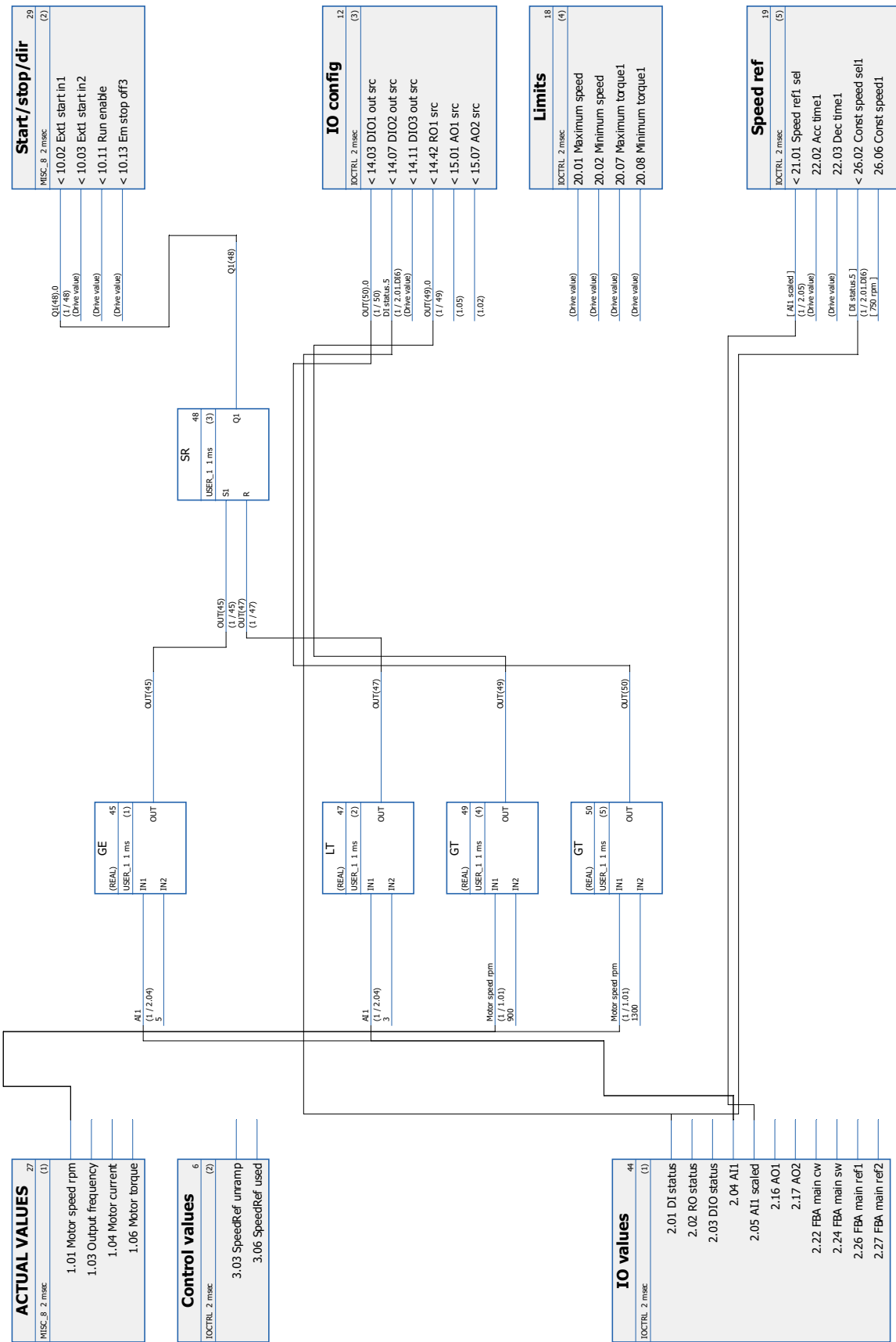
This example comprises the program presented in the previous example (page 91) as well as the following additions:

- Relay output RO1 is activated when the speed is higher than 900 rpm.
- Digital input/output DIO1 is activated when the speed is higher than 1300 rpm.
- Digital input/output DIO2 is activated when constant speed 1 (750 rpm) is activated by digital input DI6.

Additional information

- Actual signal *02.04 AI1* displays AI1 as measured.
 - Actual signal *02.01 DI status* bit 5 displays DI6.
 - Actual signal *01.01 Motor speed rpm* displays the speed.
 - The program is executed at the dedicated time level of 1 ms.
-

94 Examples of using standard function blocks

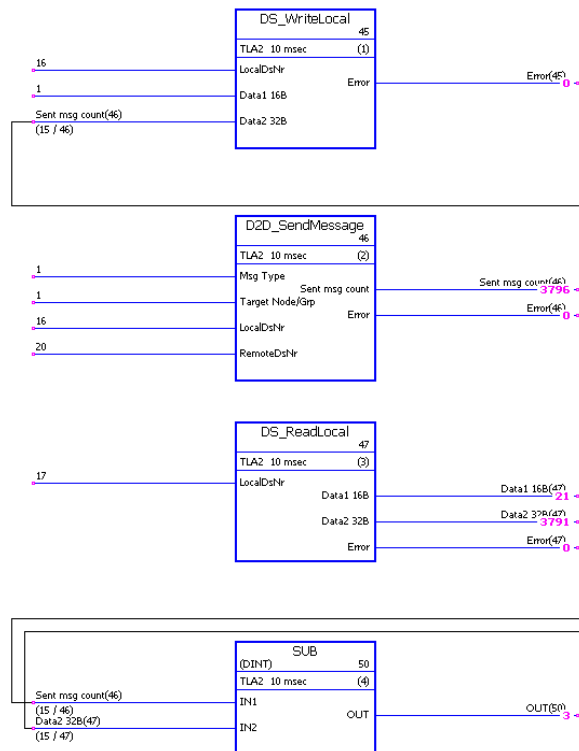


Drive-to-drive communication

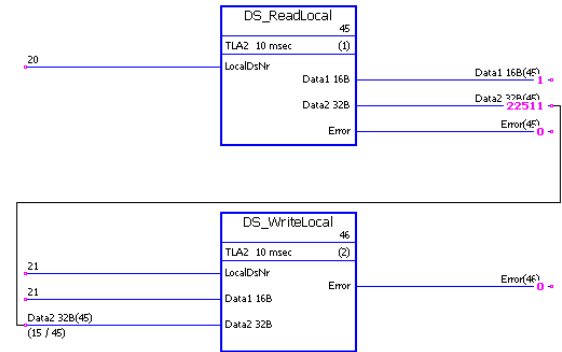
For the descriptions of the drive-to-drive standard function blocks, see section [Communication](#) on page 31.

■ Example of master point-to-point messaging

Master



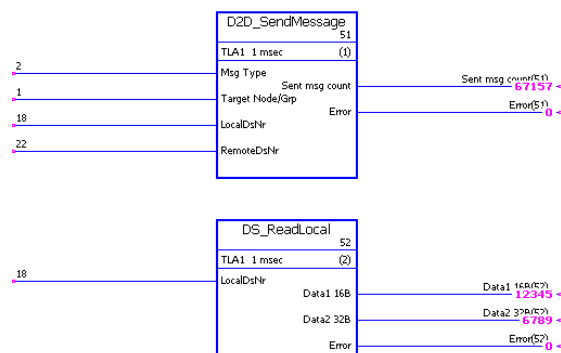
Follower (node 1)



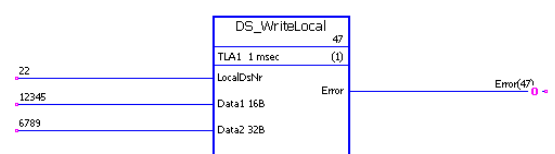
1. The master sends a constant (1) and the value of the message counter into follower dataset 20. Data is prepared to and sent from dataset 16.
2. The follower sends the received counter value and a constant (21) as a reply to the master.
3. The master calculates the difference of the latest message number and received data.

■ Example of read remote messaging

Master



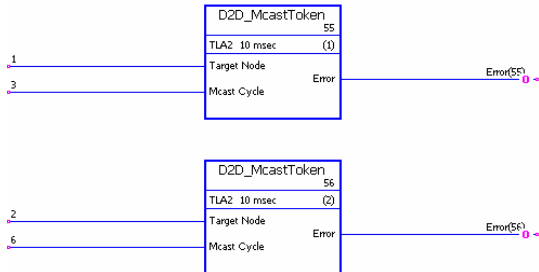
Follower (node 1)



1. The master reads the contents of the follower dataset 22 into its own dataset 18. Data is accessed using the *DS_ReadLocal* block.
2. In the follower, constant data is prepared into dataset 22.

■ Example of releasing tokens for follower-to-follower communication

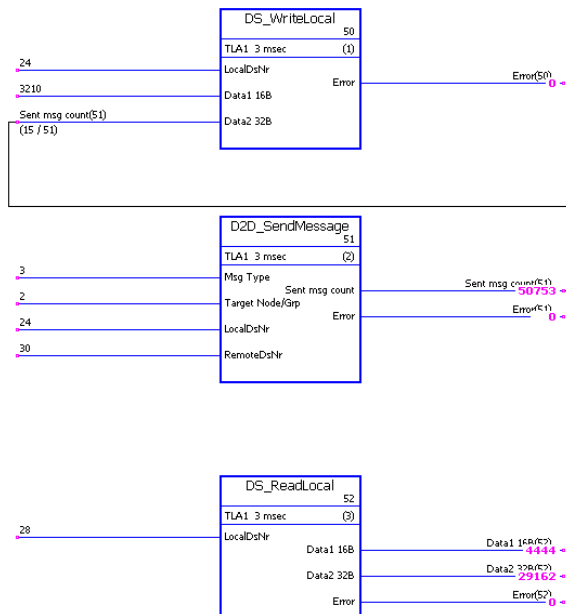
Master



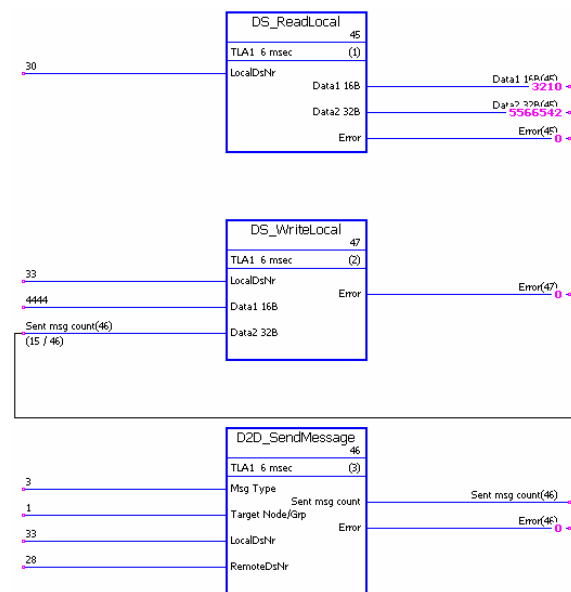
1. This drive-to-drive link consists of three drives (master and two followers).
2. The master operates as a “chairman”. Follower 1 (node 1) is allowed to send one message every 3 milliseconds. Follower 2 (node 2) is allowed to send one message every 6 milliseconds.

■ Example of follower point-to-point messaging

Follower 1 (node 1)



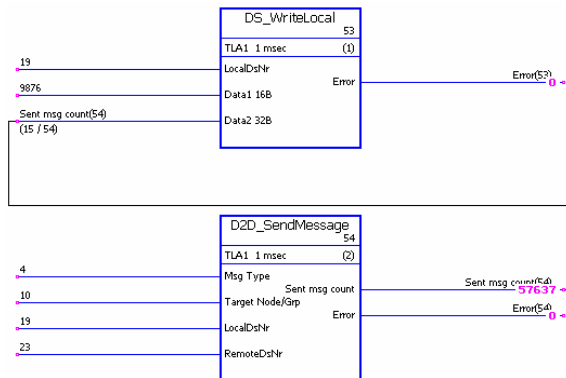
Follower 2 (node 2)



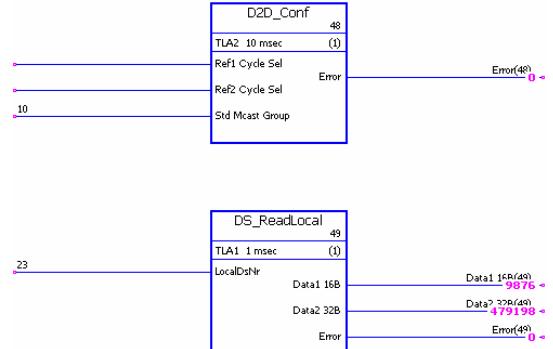
1. Follower 1 writes local dataset 24 to follower 2 dataset 30 (3 ms interval).
2. Follower 2 writes local dataset 33 to follower 1 dataset 28 (6 ms interval).
3. In addition, both followers read received data from local datasets.

■ Example of standard multicast messaging

Master



Follower(s) in Std Mcast Group 10

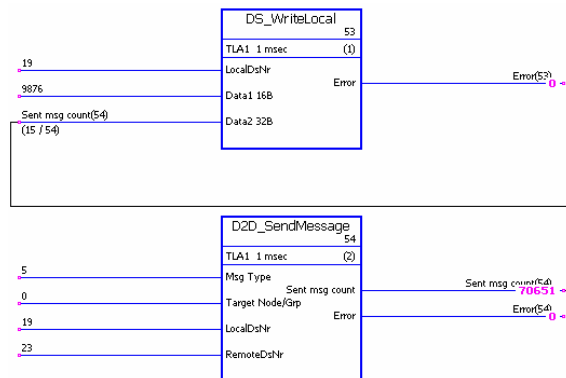


1. The master sends a constant (9876) and the value of the message counter to all followers in standard multicast group 10. The data is prepared into and sent from master dataset 19 to follower dataset 23.
2. Received data is read from dataset 23 of the receiving followers.

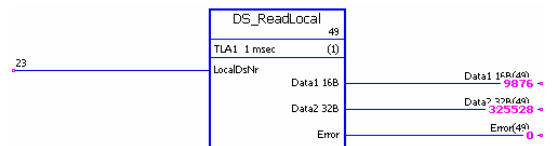
Note: The example application shown for Master above also applies to the sending follower in standard follower-to-follower multicasting.

■ Example of broadcast messaging

Master



Follower(s)



1. The master sends a constant (9876) and the value of the message counter to all followers. The data is prepared into and sent from master dataset 19 to follower dataset 23.
2. Received data is read from dataset 23 of the followers.

Note: The example application shown for Master above also applies to the sending follower in follower-to-follower broadcasting.

Further information

Product and service inquiries

Address any inquiries about the product to your local ABB representative, quoting the type designation and serial number of the unit in question. A listing of ABB sales, support and service contacts can be found by navigating to www.abb.com/drives and selecting *Sales, Support and Service network*.

Product training

For information on ABB product training, navigate to www.abb.com/drives and select *Training courses*.

Providing feedback on ABB Drives manuals

Your comments on our manuals are welcome. Go to www.abb.com/drives and select *Document Library – Manuals feedback form (LV AC drives)*.

Document library on the Internet

You can find manuals and other product documents in PDF format on the Internet. Go to www.abb.com/drives and select *Document Library*. You can browse the library or enter selection criteria, for example a document code, in the search field.

Contact us

ABB Oy

Drives
P.O. Box 184
FI-00381 HELSINKI
FINLAND
Telephone +358 10 22 11
Fax +358 10 22 22681
www.abb.com/drives

ABB Inc.

Automation Technologies
Drives & Motors
16250 West Glendale Drive
New Berlin, WI 53151
USA
Telephone 262 785-3200
1-800-HELP-365
Fax 262 780-5135
www.abb.com/drives

ABB Beijing Drive Systems Co. Ltd.

No. 1, Block D, A-10 Jiuxianqiao Beilu
Chaoyang District
Beijing, P.R. China, 100015
Telephone +86 10 5821 7788
Fax +86 10 5821 7618
www.abb.com/drives

3AUJA0000078664 Rev A EN 2010-05-26