ABB industrial drives

# Firmware manual
# Standard pump control program for ACQ810 drives

Power and productivity
for a better world™

# List of related manuals

| Drive manuals and guides | Code (English) | |
|---|---|---|
| ACQ810-04 drive modules start-up guide | *3AUA0000055159* | 2) |
| ACQ810-04 drive modules (1.1 to 45 kW, 1 to 60 hp) hardware manual | *3AUA0000055160* | 1) |
| ACQ810-04 drive modules (55 to 160 kW, 75 to 200 hp) hardware manual | *3AUA0000055161* | 1) |
| ACQ810-04 drive modules (200 to 400 kW, 250 to 600 hp) hardware manual | *3AUA0000055155* | 1) |

| Option manuals and guides | | |
|---|---|---|
| Manuals and quick guides for I/O extension modules, fieldbus adapter, etc. | | 1) |

1) Delivered in PDF format on a manuals CD with the drive module.

2) Delivered as a printed copy with the drive or optional equipment.

All manuals are available in PDF format on the Internet. See section *Further information* on the inside of the back cover.

# Firmware Manual

ACQ810 Standard Pump Control Program

Table of contents

3AUA0000055144 Rev B
EN
EFFECTIVE: 2011-02-14

# Table of contents

## *5. Application macros*

## *6. Parameters*

## *7. Additional parameter data*

## 8. Fault tracing



## 9. Control through the embedded fieldbus interface



## 10. Control through a fieldbus adapter

## *11. Control block diagrams*



## *Further information*

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 ACQ810 standard pump control program version UIFQ2000 or 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 ACQ810 control panel* provides a description and instructions for use of the control panel.
- *Control locations* describes the control locations and operation modes of the drive.
- *Program features* contains descriptions of the features of the ACQ810 standard 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 the embedded fieldbus interface.
- *Control through a fieldbus adapter* describes the communication to and from a fieldbus network using an optional fieldbus adapter module.
- *Control block diagrams* contains a graphical representation of the control program.

## 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 |
| DIO | Digital input/output; interface for digital input or output signals |
| DO | Digital output; interface for digital output signals |
| DTC | Direct torque control |
| EFB | Embedded fieldbus |
| FBA | Fieldbus adapter |
| FIO-11 | Optional analog I/O extension module |
| FIO-21 | Optional analog/digital I/O extension module |
| FIO-31 | Optional digital I/O extension module |
| FDNA-0x | Optional DeviceNet adapter |
| FENA-0x | Optional Ethernet/IP adapter |
| FLON-0x | Optional LONWORKS® adapter |
| FPBA-0x | Optional PROFIBUS DP adapter |
| FSCA-0x | Optional Modbus adapter |

| Term/abbreviation | Definition |
|---|---|
| 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 | *Power unit*; 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. |
| STO | Safe torque off |
| UIFQ xxxx | Firmware of the ACQ810 drive |
| UPS | Uninterruptible power supply; power supply equipment with battery to maintain output voltage during power failure |

2

# The ACQ810 control panel

## What this chapter contains

This chapter describes the features and operation of the ACQ810 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.

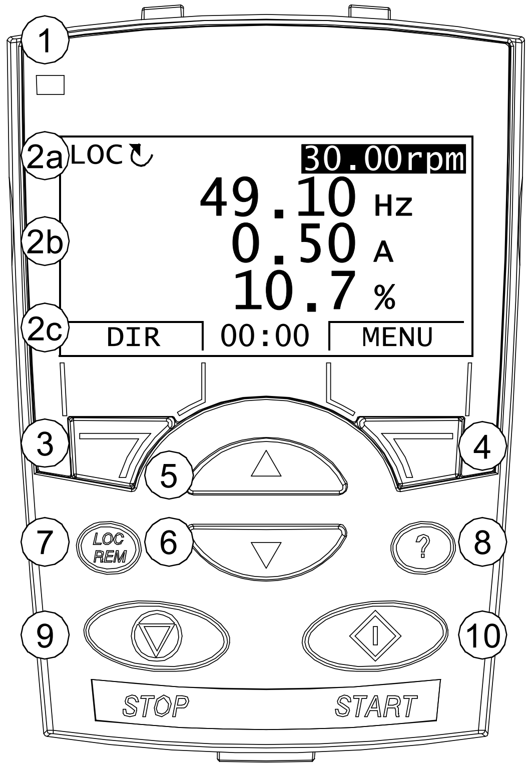
Instructions for mounting the control panel onto a cabinet door are available in *ACS-CP-U Control Panel IP54 Mounting Platform Kit Installation Guide* (3AUA0000049072 [English]).

## 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.

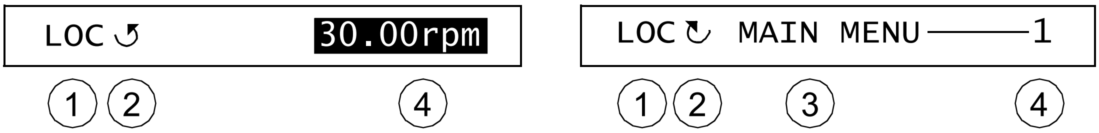
# Layout




| No. | Use |
|---|---|
| 1 | Status LED – Green for normal operation. |
| 2 | LCD display – Divided into three main areas:<br>Status line – variable, depending on the mode of operation, see section *Status line* on page *18*.<br>Center – variable; in general, shows signal and parameter values, menus or lists. Shows also faults and alarms.<br>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 –<br>Scrolls up through a menu or list displayed in the center of the LCD display.<br>Increments a value if a parameter is selected.<br>Increments the reference value if the upper right corner is highlighted.<br>Holding the key down changes the value faster. |
| 6 | Down –<br>Scrolls down through a menu or list displayed in the center of the LCD display.<br>Decrements a value if a parameter is selected.<br>Decrements the reference value if the upper right corner is highlighted.<br>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 | | • Name of the current mode<br>• Name of the list or menu shown<br>• Name of the operation state, e.g. REF EDIT. |
| 4 | Reference value or number of the selected item | | • Reference value in the Output mode<br>• Number of the highlighted item, e.g mode, parameter group or fault. |

# Operating instructions

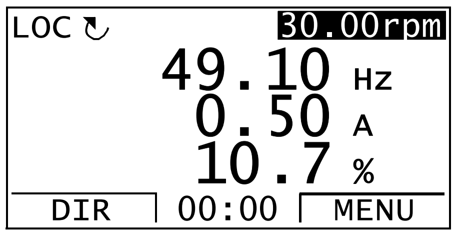
## Basics of operation

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.




## 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 or frequency 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 parameters | 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. | Press ? to read the context-sensitive help text for the item that is highlighted. | LOC ↻ TIME & DATE 6<br>TIME FORMAT<br>DATE FORMAT<br>SET TIME<br>SET DATE<br>DAYLIGHT SAVING<br>EXIT 00:00 SEL |
| | If help text exists for the item, it is shown on the display. | LOC ↻ HELP<br>Use daylight saving<br>to enable or disable<br>automatic clock<br>adjustment according<br>to daylight saving<br>EXIT 00:00 |
| 2. | If the whole text is not visible, scroll the lines with keys ▲ and ▼. | LOC ↻ HELP<br>to enable or disable<br>automatic clock<br>adjustment according<br>to daylight saving<br>changes<br>EXIT 00:00 |
| 3. | After reading the text, return to the previous display by pressing EXIT. | LOC ↻ TIME & DATE 6<br>TIME FORMAT<br>DATE FORMAT<br>SET TIME<br>SET DATE<br>DAYLIGHT SAVING<br>EXIT 00:00 SEL |

### How to find out the panel version

| Step | Action | Display |
|---|---|---|
| 1. | If the power is switched on, switch it off.<br>- If the panel cable can be disconnected easily, unplug the panel cable from the control panel, OR<br>- if the panel cable can not be disconnected easily, switch off the control board or the drive. | |
| 2. | Keep key ? depressed while you switch on the power and read the information. The display shows the following panel information:<br>Panel SW: Panel firmware version<br>ROM CRC: Panel ROM check sum<br>Flash Rev: Flash content version<br>Flash content comment.<br>When you release the ? key, the panel goes to the Output mode. | PANEL VERSION INFO<br>Panel SW: x.xx<br>Rom CRC: xxxxxxxxxx<br>Flash Rev: x.xx<br>xxxxxxxxxxxxxxxxxxxxx<br>xxxxxxxxxxxxxxxxxxxxx |

## 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. | To switch between remote control (REM shown on the status line) and local control (LOC shown on the status line), press LOC REM.<br><br>**Note:** Switching to local control can be prevented with parameter *16.01 Local lock*. | LOC ↻ MESSAGE<br>Switching to the<br>local control mode.<br><br>00:00 |
| | 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 LOC REM. The result depends on how long you press the key:<br>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*.<br>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. | |
| | To stop the drive in local control, press . | The arrow (↻ or ↺) on the status line stops rotating. |
| | To start the drive in local control, press . | The arrow (↻ or ↺) on the status line starts rotating. It is dotted until the drive reaches the setpoint. |

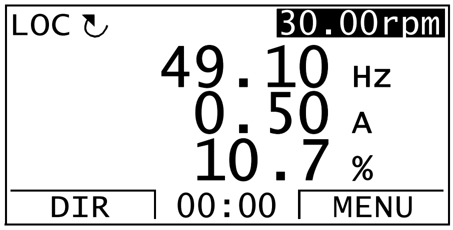
## Output mode

In the Output mode, you can:

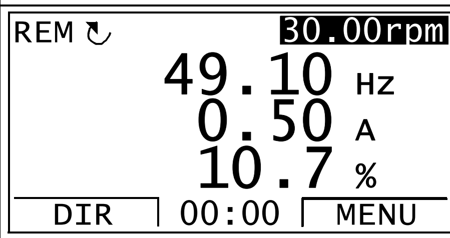
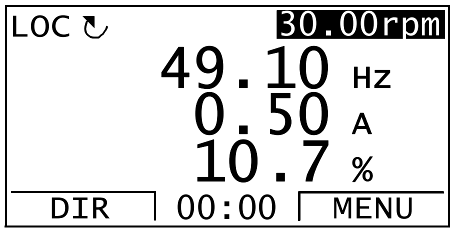
- monitor actual values of up to three signals
- change the direction of the motor rotation
- set the speed or frequency reference
- adjust the display contrast
- start, stop, change the direction and switch between local and remote control.

You get to the Output mode by pressing EXIT 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.

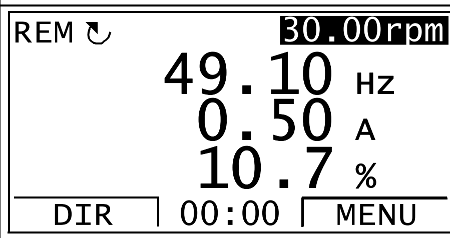



### How to change the direction of the motor rotation

| Step | Action | Display |
|---|---|---|
| 1. | If you are not in the Output mode, press EXIT 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 LOC/REM. 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 DIR. | |

### How to set the speed or frequency 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 EXIT repeatedly until you get there. |   |

| Step | Action | Display |
|---|---|---|
| 2. | If the drive is in remote control (REM shown on the status line), switch to local control by pressing LOC/REM. The display briefly shows a message about changing the mode and then returns to the Output mode. | LOC ↻ 30.00rpm<br>49.10 Hz<br>0.50 A<br>10.7 %<br>DIR 00:00 MENU |
| 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.<br>To decrease the value, press ▼. | LOC ↻ 31.00rpm<br>49.10 Hz<br>0.50 A<br>10.7 %<br>DIR 00:00 MENU |

## How to adjust the display contrast

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

## 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 MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC ↻ MAIN MENU 1<br>PARAMETERS<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |
| 2. | Go to the Parameters option by selecting PARAMETERS on the menu with keys ▲ and ▼, and pressing ENTER. | LOC ↻ PAR GROUPS 01<br>01 Actual values<br>02 I/O values<br>03 Control values<br>04 Appl values<br>05 Pump values<br>EXIT 00:00 SEL |
| 3. | Select the appropriate parameter group with keys ▲ and ▼. | LOC ↻ PAR GROUPS 99<br>99 Start-up data<br>01 Actual values<br>02 I/O values<br>03 Control values<br>04 Appl values<br>EXIT 00:00 SEL |
| | Press SEL. | LOC ↻ PARAMETERS<br>9901 Language<br>English<br>9904 Motor type<br>9905 Motor ctrl mode<br>9906 Mot nom current<br>EXIT 00:00 EDIT |
| 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 Mot nom current* is used as an example. | LOC ↻ PARAMETERS<br>9901 Language<br>9904 Motor type<br>9905 Motor ctrl mode<br>9906 Mot nom current<br>0.0 A<br>EXIT 00:00 EDIT |
| | Press EDIT. | LOC ↻ PAR EDIT<br>9906 Mot nom current<br>0.0 A<br>CANCEL 00:00 SAVE |

| Step | Action | Display |
|---|---|---|
| 5. | Specify a new value for the parameter with keys ▲ and ▼.<br>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.<br>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.<br>Pressing both keys simultaneously replaces the displayed value with the default value. | LOC ↻PAR EDIT<br>9906 Mot nom current<br>3.5 A<br>CANCEL 00:00 SAVE |
| 6. | To save the new value, press SAVE.<br>To cancel the new value and keep the original, press CANCEL. | LOC ↻ PARAMETERS<br>9906 Mot nom current<br>3.5 A<br>9907 Mot nom voltage<br>9908 Mot nom freq<br>9909 Mot nom speed<br>EXIT 00:00 EDIT |

## 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 MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC ↻ MAIN MENU 1<br>PARAMETERS<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |
| 2. | Go to the Parameters option by selecting PARAMETERS on the menu with keys ▲ and ▼, and pressing ENTER. | LOC ↻ PAR GROUPS 01<br>01 Actual values<br>02 I/O values<br>03 Control values<br>04 Appl values<br>05 Pump values<br>EXIT 00:00 SEL |
| 3. | Select the appropriate parameter group with keys ▲ and ▼. Here the value pointer parameter *21.01 Speed ref1 sel* is used as an example. | LOC ↻ PAR GROUPS 21<br>15 Analogue outputs<br>16 System<br>19 Speed calculation<br>20 Limits<br>21 Speed ref<br>EXIT 00:00 SEL |

| Step | Action | Display |
|---|---|---|
| 4. | Press SEL to select the appropriate parameter group. Select the appropriate parameter with keys ▲ and ▼, current value of each parameter is shown below it. | LOC ↻ PARAMETERS<br>2101 Speed ref1 sel<br>AI1 scaled<br>2102 Speed ref2 sel<br>2103 Speed ref1 func<br>2104 Speed ref1/2 sel<br>EXIT 00:00 EDIT |
| 5. | Press EDIT. Current value of the value pointer parameter is shown, as well as the parameter it points to. | LOC ↻ PAR EDIT<br>2101 Speed ref1 sel<br>AI1 scaled<br>[P.02.05]<br>CANCEL 00:00 SEL |
| 6. | Specify a new value with keys ▲ and ▼. The parameter the value pointer parameter points to changes respectively. | LOC ↻ PAR EDIT<br>2101 Speed ref1 sel<br>FBA ref1<br>[P.02.26]<br>CANCEL 00:00 SEL |
| 7. | Press SEL to accept any of the preselected values and to return to the parameters list.<br>The new value is shown in the parameters list. | LOC ↻ PARAMETERS<br>2101 Speed ref1 sel<br>FBA ref1<br>2102 Speed ref2 sel<br>2105 Speed share<br>2109 SpeedRef min abs<br>EXIT 00:00 EDIT |
| | To freely define an analog signal as the value, choose Pointer and press NEXT. The parameter group and index will be shown.<br>Select the parameter group with ▲ and ▼. The text below the cursor displays the currently-selected parameter group. | LOC ↻ PAR EDIT<br>2101 Speed ref1 sel<br>P.02.05<br>02 I/O values<br>CANCEL 00:00 SAVE |
| 8. | Press NEXT to select the parameter index.<br>Again, the text below the cursor reflects the current setting. | LOC ↻ PAR EDIT<br>2101 Speed ref1 sel<br>P.02.07<br>0207 AI2 scaled<br>CANCEL 00:00 SAVE |
| 9. | To save the new value for the pointer parameter, press SAVE.<br>The new value is shown in the parameters list. | LOC ↻ PARAMETERS<br>2101 Speed ref1 sel<br>AI2 scaled<br>2102 Speed ref2 sel<br>2105 Speed share<br>2109 SpeedRef min abs<br>EXIT 00:00 EDIT |

### 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 MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC ↻ MAIN MENU 1<br>**PARAMETERS**<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |
| 2. | Go to the Parameters option by selecting PARAMETERS on the menu with keys ▲ and ▼, and pressing ENTER. | LOC ↻ PAR GROUPS 01<br>**01 Actual values**<br>02 I/O values<br>03 Control values<br>04 Appl values<br>05 Pump values<br>EXIT 00:00 SEL |
| 3. | Select the appropriate parameter group with keys ▲ and ▼. Here the bit pointer parameter *10.02 Ext1 start in1* is used as an example. | LOC ↻ PAR GROUPS 10<br>**10 Start/stop/dir**<br>11 Start/stop mode<br>12 Operating mode<br>13 Analogue inputs<br>14 Digital I/O<br>EXIT 00:00 SEL |
| 4. | Press SEL to select the appropriate parameter group. Current value of each parameter is shown below its name. | LOC ↻ PARAMETERS<br>**1001 Ext1 start func**<br>**In1**<br>1002 Ext1 start in1<br>1003 Ext1 start in2<br>1004 Ext2 start func<br>EXIT 00:00 EDIT |
| | Select the parameter *10.02 Ext1 start in1* with keys ▲ and ▼. | LOC ↻ PARAMETERS<br>1001 Ext1 start func<br>**1002 Ext1 start in1**<br>**DI1**<br>1003 Ext1 start in2<br>1004 Ext2 start func<br>EXIT 00:00 EDIT |
| 5. | Press EDIT. | LOC ↻ PAR EDIT<br>1002 Ext1 start in1<br>DI1<br>[P.02.01.00]<br>CANCEL 00:00 SEL |

| Step | Action | Display |
|---|---|---|
| 6. | Specify a new value with keys ▲ and ▼. The text below the cursor shows the corresponding parameter group, index and bit. | LOC ↻ PAR EDIT<br>1002 Ext1 start in1<br>DIO4<br>[P.02.03.03]<br>CANCEL 00:00 SEL |
| 7. | Press SEL to accept any of the preselected values and to return to the parameters list. | LOC ↻ PARAMETERS<br>1002 Ext1 start in1<br>DIO4<br>1003 Ext1 start in2<br>1004 Ext2 start func<br>1005 Ext2 start in1<br>EXIT 00:00 EDIT |
| | To freely define a bit of a binary parameter as the value, choose Pointer and press NEXT. The parameter group, index and bit will be shown.<br>Select the parameter group with ▲ and ▼. The text below the cursor displays the currently-selected parameter group. | LOC ↻ PAR EDIT<br>1002 Ext1 start in1<br>P.02.01.00<br>02 I/O values<br>CANCEL 00:00 SAVE |
| 8. | Press NEXT to select the parameter index.<br>Again, the text below the cursor reflects the current setting. | LOC ↻ PAR EDIT<br>1002 Ext1 start in1<br>P.02.01.00<br>0201 DI status<br>CANCEL 00:00 SAVE |
| 9. | Press NEXT to select the bit.<br>Again, the text below the cursor reflects the current setting. | LOC ↻ PAR EDIT<br>1002 Ext1 start in1<br>P.02.01.01<br>01 DI2<br>CANCEL 00:00 SAVE |
| 10. | To save the new value for the pointer parameter, press SAVE .<br>The new value is shown in the parameters list. | LOC ↻ PARAMETERS<br>1002 Ext1 start in1<br>P.02.01.01<br>1003 Ext1 start in2<br>1004 Ext2 start func<br>1005 Ext2 start in1<br>EXIT 00:00 EDIT |

### 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 MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC ↻ MAIN MENU 1<br>**PARAMETERS**<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |
| 2. | Go to the Parameters option by selecting PARAMETERS on the menu with keys ▲ and ▼, and pressing ENTER. | LOC ↻ PAR GROUPS 01<br>**01 Actual values**<br>02 I/O values<br>03 Control values<br>04 Appl values<br>05 Pump values<br>EXIT 00:00 SEL |
| | Select the appropriate parameter group with keys ▲ and ▼. Here the bit pointer parameter *14.07 DIO2 out src* is used as an example. | LOC ↻ PAR GROUPS 14<br>10 Start/stop/dir<br>11 Start/stop mode<br>12 Operating mode<br>13 Analogue inputs<br>**14 Digital I/O**<br>EXIT 00:00 SEL |
| 3. | Press SEL to select the appropriate parameter group. Select the appropriate parameter with keys ▲ and ▼. Current value of each parameter is shown below its name. | LOC ↻ PARAMETERS<br>1404 DIO1 Ton<br>1405 DIO1 Toff<br>1406 DIO2 conf<br>**1407 DIO2 out src**<br>**P.06.02.03**<br>EXIT 00:00 EDIT |
| 4. | Press EDIT. | LOC ↻ PAR EDIT<br>1407 DIO2 out src<br>Pointer<br>CANCEL 00:00 NEXT |
| | Select CONST with keys ▲ and ▼. | LOC ↻ PAR EDIT<br>1407 DIO2 out src<br>Const<br>CANCEL 00:00 NEXT |

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

## How to select the monitored signals

| Step | Action | Display |
|---|---|---|
| 1. | 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.<br>**Note:** If you set one of the parameters *56.01*…*56.03* 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 *56.04*…*56.06* to *Disabled*. | LOC ↻ PAR EDIT<br>5601 Signal1 param<br>01.03<br>CANCEL 00:00 NEXT<br><br>LOC ↻ PAR EDIT<br>5602 Signal2 param<br>01.04<br>CANCEL 00:00 NEXT<br><br>LOC ↻ PAR EDIT<br>5603 Signal3 param<br>01.06<br>CANCEL 00:00 NEXT |

## 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.

An assistant may merely consist of a sequence of parameters that the user must adjust, but may also involve questions; based on the answers, one or several parameters are automatically adjusted. The assistant may also display additional information about the selections.

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.

Different firmware versions may include different assistants.

### How to invoke an assistant

The table below shows how assistants are invoked.

| Step | Action | Display |
|---|---|---|
| 1. | Go to the Main menu by pressing MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC MAIN MENU 1<br>PARAMETERS<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |
| 2. | Go to the Assistants mode by selecting ASSISTANTS on the menu with keys ▲ and ▼, and pressing ENTER. | LOC CHOICE 1/3<br>Select assistant:<br>Motor Set-up<br>Application Macro<br>Start-up assistant<br>EXIT 00:00 OK |
| 3. | Select an assistant by highlighting it with ▲ and ▼, and by pressing OK.<br>• Whenever the assistant prompts the user to adjust a parameter, the adjustment is made as described starting on page 25.<br>• Whenever the assistant prompts a question, select the most suitable answer with ▲ and ▼, and press OK. | LOC PAR EDIT<br>9904 Motor type<br>AM<br>[0]<br>EXIT 00:00 SAVE |
| 4. | After the assistant has been completed, the main menu is displayed. To run another assistant, repeat the procedure from step 2.<br>To abort the assistant at any point, press EXIT. | |

## 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 MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC ↻MAIN MENU 1<br>PARAMETERS<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |
| 2. | Go to the Changed Parameters mode by selecting CHANGED PAR on the menu with keys ▲ and ▼, and pressing ENTER.<br>If there are no changed parameters in the history, corresponding text will be shown. | LOC ↻ MESSAGE<br>No parameters<br><br>00:00 |
| | 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. | LOC ↻CHANGED PAR<br>9906 Mot nom current<br>3.5 A<br>9907 Mot nom voltage<br>9908 Mot nom freq<br>9909 Mot nom speed<br>EXIT 00:00 EDIT |
| 3. | Press EDIT to modify the value. | LOC ↻PAR EDIT<br>9906 Mot nom current<br>3.5 A<br>CANCEL 00:00 SAVE |
| 4. | Specify a new value for the parameter with keys ▲ and ▼.<br>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. | LOC ↻PAR EDIT<br>9906 Mot nom current<br>3.0 A<br>CANCEL 00:00 SAVE |

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

## 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 MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC ↻ MAIN MENU 1<br>PARAMETERS<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |
| 2. | Go to the Fault Logger option by selecting FAULT LOGGER on the menu with keys ▲ and ▼, and pressing ENTER.<br>If there are no faults in the fault history, corresponding text will be shown. | LOC ↻ MESSAGE<br>No fault history<br>found |
| | 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 *313*). | LOC ↻ FAULT LOGGER 1<br>36: LOCAL CTRL LOSS<br>29.04.08 10:45:58<br>EXIT 00:00 DETAIL |
| 3. | To see the details of a fault, select it with keys ▲ and ▼, and press DETAIL.<br>Scroll the text with keys ▲ and ▼.<br>To return to the previous display, press EXIT. | LOC ↻ LOCAL CTRL LOSS<br>TIME<br>10:45:58<br>FAULT CODE<br>36<br>FAULT CODE EXTENSION<br>EXIT 00:00 DIAG |
| 4. | If you want help in diagnosing the fault, press DIAG. | LOC ↻<br>Check parameter '30.0<br>3 Local ctrl loss' se<br>tting. Check PC tool<br>or panel connection.<br>EXIT OK |

| Step | Action | Display |
|---|---|---|
| 5. | Press OK. The panel allows you to edit necessary parameters to correct the fault. | LOC ↻ PAR EDIT<br>3003 Local ctrl loss<br>Fault<br>[1]<br>EXIT 00:00 SAVE |
| 6. | Specify a new value for the parameter with keys ▲ and ▼.<br>To accept the new value, press SAVE.<br>To cancel the new value and keep the original, press EXIT. | LOC ↻ PAR EDIT<br>3003 Local ctrl loss<br>Spd ref Safe<br>[2]<br>EXIT 00:00 SAVE |

## How to reset faults

| Step | Action | Display |
|---|---|---|
| 1. | When a fault occurs, a text identifying the fault is shown.<br>To reset the fault, press RESET.<br>To return to the previous display, press EXIT. | LOC ↻ FAULT<br>FAULT 36<br>LOCAL CTRL LOSS<br>RESET EXIT |

## 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 MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC ↻ MAIN MENU 1<br>PARAMETERS<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |
| 2. | Go to the Time & Date option by selecting TIME & DATE on the menu with keys ▲ and ▼, and pressing ENTER. | LOC ↻ TIME & DATE 1<br>CLOCK VISIBILITY<br>TIME FORMAT<br>DATE FORMAT<br>SET TIME<br>SET DATE<br>EXIT 00:00 SEL |
| 3. | To show (hide) the clock, select CLOCK VISIBILITY on the menu, press SEL, select Show clock (Hide clock) with keys ▲ and ▼ and press SEL, or, if you want to return to the previous display without making changes, press EXIT. | LOC ↻ CLOCK VISIB 1<br>Show clock<br>Hide clock<br>EXIT 00:00 SEL |
| | To specify the time format, select TIME FORMAT on the menu, press SEL and select a suitable format with keys ▲ and ▼. Press SEL to save or CANCEL to cancel your changes. | LOC ↻ TIME FORMAT 1<br>24-hour<br>12-hour<br>CANCEL 00:00 SEL |

| Step | Action | Display |
|---|---|---|
| | To specify the date format, select DATE FORMAT on the menu, press SEL and select a suitable format.<br>Press OK to save or CANCEL to cancel your changes. | LOC ↻ DATE FORMAT 1<br>dd.mm.yy<br>mm/dd/yy<br>dd.mm.yyyy<br>mm/dd/yyyy<br>CANCEL 00:00 OK |
| | To set the time, select SET TIME on the menu and press SEL.<br>Specify the hours with keys ▲ and ▼, and press OK.<br>Then specify the minutes. Press OK to save or CANCEL to cancel your changes. | LOC ↻ SET TIME<br>15:41<br>CANCEL OK |
| | To set the date, select SET DATE on the menu and press SEL.<br>Specify the first part of the date (day or month depending on the selected date format) with keys ▲ and ▼, and press OK. Repeat for the second part.<br>After specifying the year, press OK. To cancel your changes, press CANCEL. | LOC ↻ SET DATE<br>19.03.2008<br>CANCEL 00:00 OK |
| | To enable or disable the automatic clock transitions according to the daylight saving changes, select DAYLIGHT SAVING on the menu and press SEL.<br>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 EXIT.<br>To disable automatic clock transitions according to the daylight saving changes, select Off and press SEL.<br>To enable automatic clock transitions, select the country or area whose daylight saving changes are followed and press SEL.<br>To return to the previous display without making changes, press EXIT. | LOC ↻ DAYLIGHT SAV 1<br>Off<br>EU<br>US<br>Australia1:NSW,Vict..<br>Australia2:Tasmania..<br>EXIT 00:00 SEL<br><br>LOC ↻ HELP<br>EU:<br>On: Mar last Sunday<br>Off: Oct last Sunday<br><br>US:<br>EXIT 00:00 |

## 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 MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC ↻ MAIN MENU 1<br>PARAMETERS<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |

| Step | Action | Display |
|---|---|---|
| 2. | Go to the Parameter Backup option by selecting PAR BACKUP on the menu with keys and , and pressing ENTER . | LOC ↻ PAR BACKUP——1<br>MAKE BACKUP TO PANEL<br>SHOW BACKUP INFO<br>RESTORE PARS ALL<br>RESTORE PARS NO-IDRUN<br>RESTORE PARS IDRUN<br>EXIT 00:00 SEL |
| | 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 SEL . Operation starts. Press ABORT if you want to stop the operation. | LOC ↻ PAR BACKUP<br>Copying file 1/2<br>ABORT 00:00 |
| | After the backup is completed, the display shows a message about the completion. Press OK to return to the Par Backup. | LOC ↻ MESSAGE<br>Parameter upload successful<br>OK 00:00 |
| | To perform restore functions, select the appropriate operation (here RESTORE PARS ALL is used as an example) on the Par Backup with keys and . | LOC ↻ PAR BACKUP——3<br>MAKE BACKUP TO PANEL<br>SHOW BACKUP INFO<br>RESTORE PARS ALL<br>RESTORE PARS NO-IDRUN<br>RESTORE PARS IDRUN<br>EXIT 00:00 SEL |
| | Press SEL . Restoring starts. | LOC ↻ PAR BACKUP<br>Initializing param restore operation<br>00:00 |
| | A version check is made. Scroll the text with keys and . | LOC ↻ VERSION CHECK -1<br>FIRMWARE VERSION<br>UIFQ,200F,0,<br>UIFQ,200C,0,<br>OK<br>PRODUCT VARIANT<br>CANCEL 00:00 CONT |
| | If you want to continue, press CONT . Press CANCEL if you want to stop the operation. If the downloading is continued, the display shows a message about it. | LOC ↻ PAR BACKUP<br>Initializing param. restore operation<br>00:00 |

| Step | Action | Display |
|---|---|---|
| | Downloading continues, drive is being restarted. | LOC ↻ PAR BACKUP<br>Restarting drive<br>00:00 |
| | The display shows the transfer status as a percentage of completion. | LOC ↻ PAR BACKUP<br>Restoring/downloading<br>all parameters<br>50% |
| | Downloading finishes. | LOC ↻ PAR BACKUP<br>Finishing restore<br>operation |

### 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. | LOC ↻ PAR BACKUP<br>Initializing param.<br>restore operation<br>00:00 |
| 2. | A version check is made.<br>You can see on the panel that the firmware versions are not the same. | LOC ↻ VERSION CHECK -1<br>FIRMWARE VERSION<br>UIFQ,200F,0,<br>UIFQ,200C,0,<br>OK<br>PRODUCT VARIANT<br>CANCEL 00:00 CONT |
| | Scroll the text with keys ▲ and ▼.<br>To continue, press CONT. Press CANCEL to stop the operation. | LOC ↻ VER CHECK -2<br>FIRMWARE VERSION<br>PRODUCT VARIANT<br>7<br>7<br>OK<br>CANCEL 00:00 CONT |

| Step | Action | Display |
|---|---|---|
| 3. | If the downloading is continued, the display shows a message about it. | LOC ↻ PAR BACKUP<br>Initializing param<br>restore operation<br>00:00 |
| | Downloading continues, drive is being restarted. | LOC ↻ PAR BACKUP<br>Restarting drive<br>00:00 |
| | The display shows the transfer status as a percentage of completion. | LOC ↻ PAR BACKUP<br>Restoring/downloading<br>all parameters<br>50% |
| | Downloading continues. | LOC ↻ PAR BACKUP<br>Restarting drive<br>00:00 |
| | Downloading finishes. | LOC ↻ PAR BACKUP<br>Finishing restore<br>operation |
| 4. | The panel shows a list of erroneous parameters. | LOC ↻ PARAM ERRORS— 1<br>9401 Ext IO1 sel<br>0<br>?<br>INCORRECT VALUE TYPE<br>9402 Ext IO2 sel<br>READY 00:00 |
| | You can scroll the parameters with keys ▲ and ▼. The reason for parameter error is also shown. | LOC ↻ PARAM ERRORS—13<br>21110<br>21201<br>1<br>?<br>PARAMETER NOT FOUND<br>READY 00:00 |

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

## 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. | LOC ↻ PAR BACKUP<br>Initializing param<br>restore operation<br>00:00 |
| 2. | Version check is also OK.<br>You can see on the panel that the firmware versions are not the same.<br><br>You can scroll the text with keys ▲ and ▼. | LOC ↻ VER CHECK 1<br>FIRMWARE VERSION<br>UIFQ,200F,0,<br>UIFQ,200C,0,<br>OK<br>PRODUCT VARIANT<br>CANCEL 00:00 CONT<br><br>LOC ↻ VER CHECK 2<br>FIRMWARE VERSION<br>PRODUCT VARIANT<br>7<br>7<br>OK<br>CANCEL 00:00 CONT |

| Step | Action | Display |
| --- | --- | --- |
| 3. | If the downloading is continued, the display shows a message about it. | LOC ↻ PAR BACKUP<br>Initializing param restore operation<br>00:00 |
| 4. | Downloading continues, drive is being restarted. | LOC ↻ PAR BACKUP<br>Restarting drive<br>00:00 |
| 5. | The display shows the transfer status as a percentage of completion. | LOC ↻ PAR BACKUP<br>Restoring/downloading user set 1<br>50% |
| 6. | Downloading continues. | LOC ↻ PAR BACKUP<br>Initializing param restore operation<br>00:00 |
| 7. | Downloading continues, drive is being restarted. | LOC ↻ PAR BACKUP<br>Restarting drive<br>00:00 |
| 8. | Downloading finishes. | LOC ↻ PAR BACKUP<br>Finishing restore operation |
| 9. | Panel shows a text identifying the alarm and returns to the Par Backup. | LOC ↻ ALARM<br>ALARM 2036<br>RESTORE<br>EXIT |

## 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*.<br>A user set is loaded through parameter *16.09 User set sel*. Select parameter group *16 System* with keys ▲ and ▼. | LOC ↻ PAR GROUPS 16<br>12 Operating mode<br>13 Analogue inputs<br>14 Digital I/O<br>15 Analogue outputs<br>16 System<br>EXIT 00:00 SEL |
| 2. | Press SEL 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. | LOC ↻ PARAMETERS<br>1603 Pass code<br>1604 Param restore<br>1607 Param save<br>1609 User set sel<br>No request<br>EXIT 00:00 EDIT |
| 3. | Press EDIT. | LOC ↻ PAR EDIT<br>1609 User set sel<br>No request<br>[1]<br>CANCEL 00:00 SAVE |
| | Select the user set you want to load with keys ▲ and ▼.<br>Press SAVE. | LOC ↻ PAR EDIT<br>1609 User set sel<br>Load set 1<br>[2]<br>CANCEL 00:00 SAVE |
| 4. | Panel shows a text identifying the fault. | LOC ↻ FAULT<br>FAULT 310<br>USERSET LOAD<br>RESET EXIT |

## How to view information about the backup

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

| Step | Action | Display |
|---|---|---|
| 2. | Go to the Par Backup option by selecting PAR BACKUP on the menu with keys ▲ and ▼, and pressing ENTER.<br>Select SHOW BACKUP INFO with keys ▲ and ▼. | LOC ↻ PAR BACKUP——2<br>MAKE BACKUP TO PANEL<br>SHOW BACKUP INFO<br>RESTORE PARS ALL<br>RESTORE PARS NO-IDRUN<br>RESTORE PARS IDRUN<br>EXIT 00:00 SEL |
| 3. | Press SEL. The display shows the following information about the drive from where the backup was made:<br>BACKUP INTERFACE VER: Format version of the backup file<br>FIRMWARE VERSION: Information on the firmware<br>UIFQ: Firmware of the ACQ810 drive<br>2010: Firmware version<br>0: Firmware patch version<br>PRODUCT VARIANT:<br>7: ACQ810 (Pump control program)<br>You can scroll the information with ▲ and ▼. | LOC ↻ BACKUP INFO——<br>BACKUP INTERFACE VER<br>0.4<br>0.4<br>FIRMWARE VERSION<br>UIFQ,2010,0,<br>EXIT 00:00<br><br>LOC ↻ BACKUP INFO——<br>FIRMWARE VERSION<br>UIFQ,2010,0,<br>UIFQ,200F,0,<br>PRODUCT VARIANT<br>7<br>EXIT 00:00 |
| 4. | Press EXIT to return to the Par Backup. | LOC ↻ PAR BACKUP——1<br>MAKE BACKUP TO PANEL<br>SHOW BACKUP INFO<br>RESTORE PARS ALL<br>RESTORE PARS NO-IDRUN<br>RESTORE PARS IDRUN<br>EXIT 00:00 SEL |

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

| 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). | LOC ↻I/O SETTINGS—1<br>DI1<br>1002 Ext1 start in1<br>DI2<br>DI3<br>1010 Fault reset sel<br>EXIT 00:00 EDIT |
| 6. | Press EDIT. | LOC ↻PAR EDIT<br>1002 Ext1 start in1<br>DI1<br>[P.02.01.00]<br>CANCEL 00:00 SEL |
| 7. | Specify a new value for the setting with keys ▲ and ▼.<br>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. | LOC ↻PAR EDIT<br>1002 Ext1 start in1<br>DI04<br>[P.02.03.03]<br>CANCEL 00:00 SEL |
| 8. | To save the new value, press SEL.<br>To cancel the new value and keep the original, press CANCEL. | LOC ↻I/O SETTINGS—1<br>DI1<br>1002 Ext1 start in1<br>DI2<br>DI3<br>1010 Fault reset sel<br>EXIT 00:00 EDIT |

## 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 LOC/REM. Reference editing is not possible in remote control mode. (See page 22 for more information on switching between the local and remote control modes.)<br>The display shows a message about that if you try to enter REF EDIT in the remote control mode. | REM ↻ MESSAGE<br>Reference editing<br>enabled only in<br>local control mode<br>00:00 |
| 2. | Otherwise, go to the Main menu by pressing MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC ↻MAIN MENU 1<br>PARAMETERS<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |
| 3. | Go to the Reference Edit option by selecting REF EDIT on the menu with keys ▲ and ▼, and pressing ENTER. | LOC ↻ REF EDIT<br>█0000.00rpm<br>CANCEL 00:00 NEXT |
| 4. | Select the correct sign with keys ▲ and ▼, and press NEXT. Select the correct numbers with keys ▲ and ▼, and after each number is selected, press NEXT. | LOC ↻ REF EDIT<br>-1250.00rpm<br>CANCEL 00:00 SAVE |
| 5. | After the last number is selected, press SAVE. Go to the Output mode by pressing EXIT. The selected reference value is shown in the status line. | LOC ↻ -1250.00rpm<br>49.10 Hz<br>0.50 A<br>10.7 %<br>DIR 00:00 MENU |

## 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 MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC ↻MAIN MENU 1<br>PARAMETERS<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |
| 2. | Go to the Drive info option by selecting DRIVE INFO on the menu with keys ▲ and ▼, and pressing ENTER. | LOC ↻ DRIVE INFO<br>DRIVE NAME<br>-<br>DRIVE TYPE<br>ACQ810<br>DRIVE MODEL<br>EXIT 00:00 |
| 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.<br>DRIVE NAME: Drive name defined as a text in DriveStudio commissioning and maintenance tool<br>DRIVE TYPE: e.g. ACQ810<br>DRIVE MODEL: Type code of the drive<br>FW VERSION: See page 45.<br>SOLUTION PROGRAM: Version information of the active application program<br>BASE SOLUTION PROGRAM: Version information of the application program template<br>STANDARD LIBRARY: Version information of the standard library<br>TECHNOLOGY LIBRARY: Not applicable to the ACQ810<br>POWER UNIT SERNO: Serial number of the power stage (JPU)<br>MEM UNIT HW SERNO: Serial number in manufacturing the memory unit (JMU)<br>MEM UNIT CONFIG SERNO: Serial number in configuring the memory unit (JMU).<br>Press EXIT to return to the Main menu. | LOC ↻ DRIVE INFO<br>FW VERSION<br>UIFQ,2010,0,<br>SOLUTION PROGRAM<br>-<br>BASE SOLUTION PROGRAM<br>EXIT 00:00 |

## 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 MENU if you are in the Output mode.<br>Otherwise press EXIT repeatedly until you get to the Main menu. | LOC ↻MAIN MENU 1<br>PARAMETERS<br>ASSISTANTS<br>CHANGED PAR<br>EXIT 00:00 ENTER |
| 2. | Go to the Parameter Change Log option by selecting PAR CHG LOG on the menu with keys ▲ and ▼, and pressing ENTER.<br>If there are no parameter changes in the history, corresponding text will be shown. | LOC ↻ MESSAGE<br>No parameters<br>available<br>00:00 |
| | 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 ▼. | LOC ↻ LAST CHANGES 1<br>9402 Ext IO2 sel<br>None<br>11.09.2008 12:04:55<br>9401 Ext IO1 sel<br>9402 Ext IO2 sel<br>EXIT 00:00 EDIT |
| 3. | If you want to edit a parameter, select the parameter with keys ▲ and ▼ and press EDIT. | LOC ↻ PAR EDIT<br>9402 Ext IO2 sel<br>None<br>[0]<br>CANCEL 00:00 SAVE |
| 4. | Specify a new value for the parameter with keys ▲ and ▼.<br>To save the new value, press SAVE.<br>To cancel the new value and keep the original, press CANCEL. | LOC ↻ PAR EDIT<br>9402 Ext IO2 sel<br>FIO-01<br>[1]<br>CANCEL 00:00 SAVE |

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

3

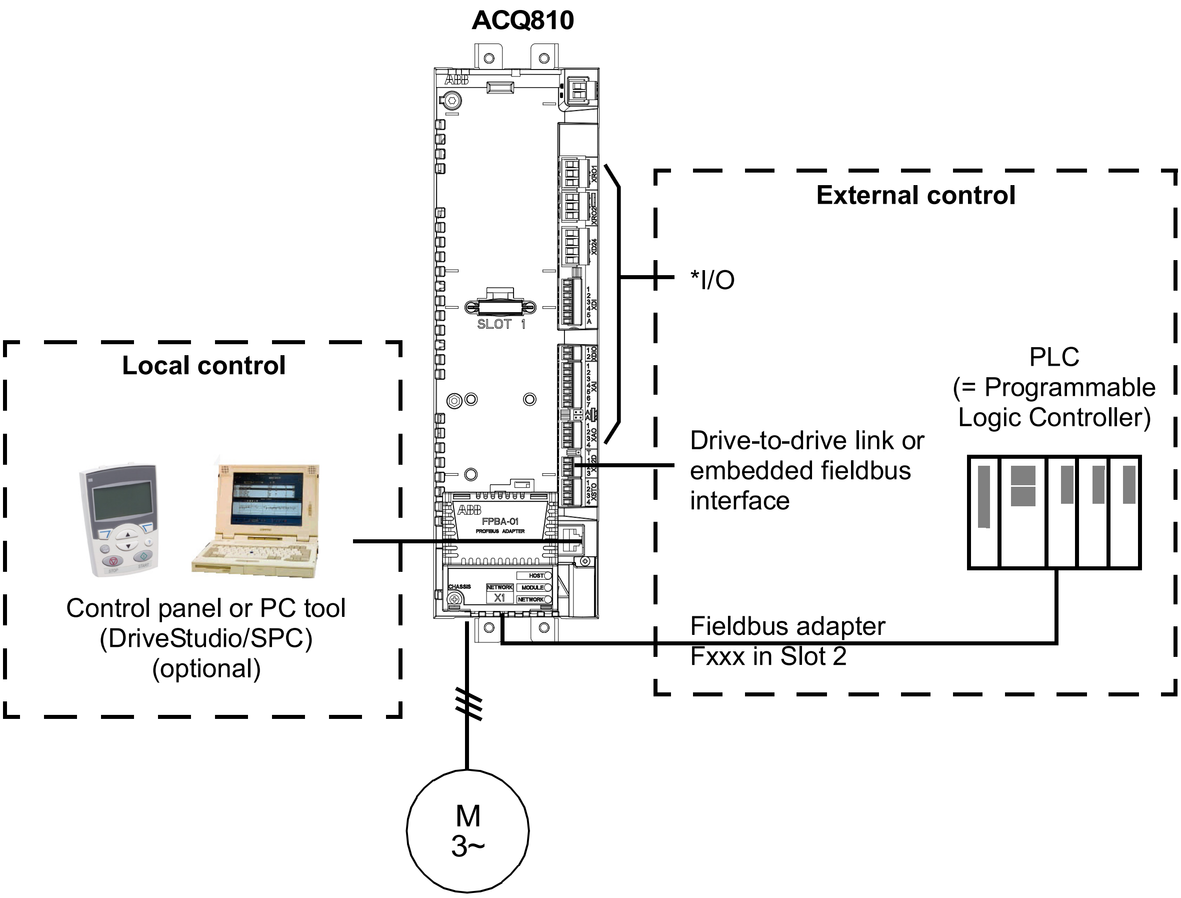
# Control locations

## What this chapter contains

This chapter describes the control locations of the drive.

# 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).




*Extra inputs/outputs can be added by installing an optional I/O extension module (FIO-xx) into drive Slot 1.

## 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. A speed control mode is 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 the embedded fieldbus interface or an optional fieldbus adapter module), the I/O terminals (digital and analog inputs), optional I/O extension modules or the drive-to-drive link. External references are given through the fieldbus interface, analog inputs, or drive-to-drive link.

Two external control locations, EXT1 and EXT2, are available. The user can select control signals (e.g. start and stop) and control modes separately 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 signal or fieldbus control word.

4

# Program features

## What this chapter contains

This chapter describes the features of the control program.

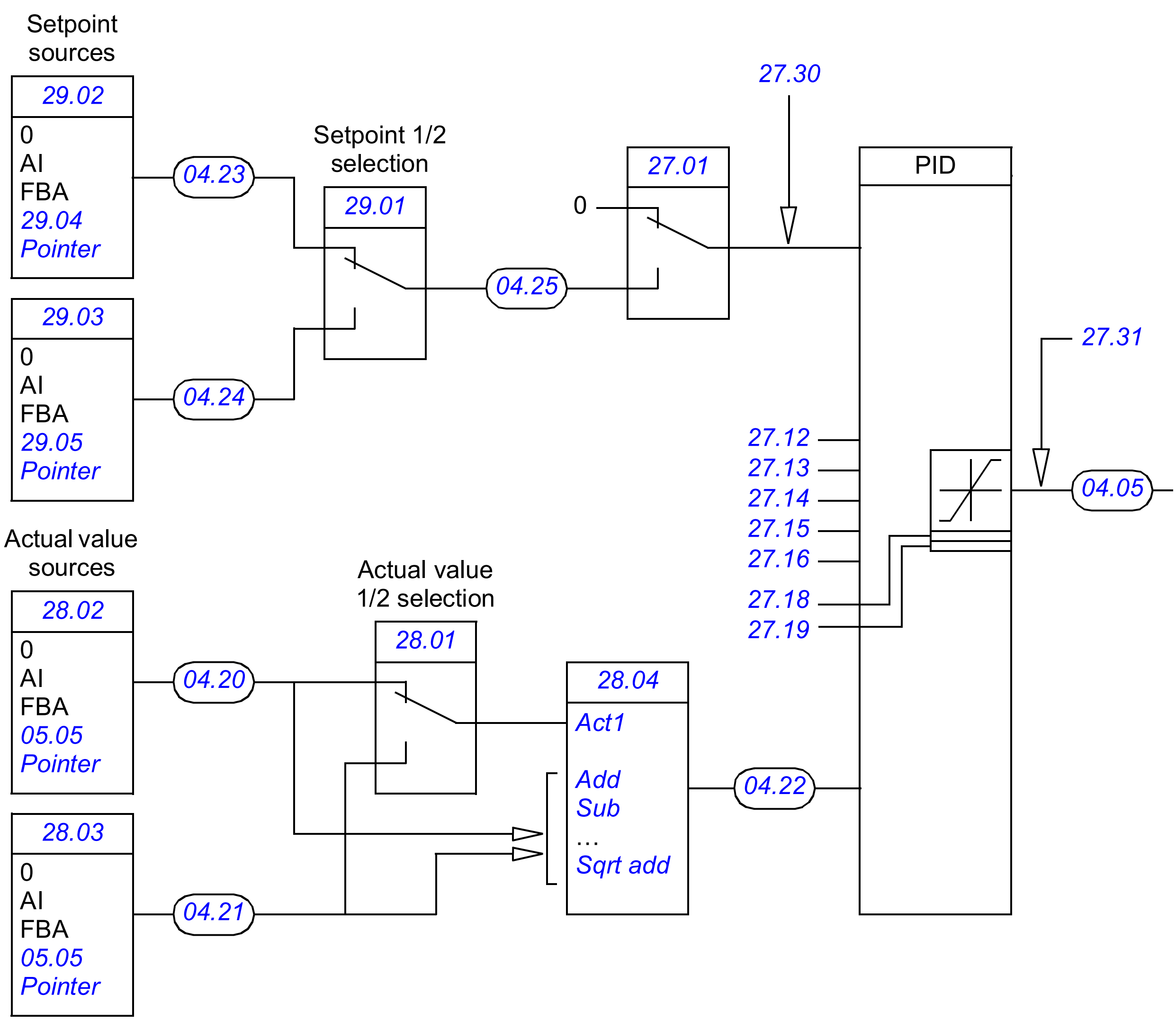
# Pump control features

## 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 setpoint (reference) is connected to the drive instead of a speed reference. A process actual value (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 control program allows switching between two different setpoints and actual values.

The simplified block diagram below illustrates the process PID control. For a more detailed diagram, see page *372*.




### Settings

Parameter groups *12 Operating mode* (page *141*), *27 Process PID* (page *187*), *28 Procact sel* (page *191*), and *29 Setpoint sel* (page *193*).

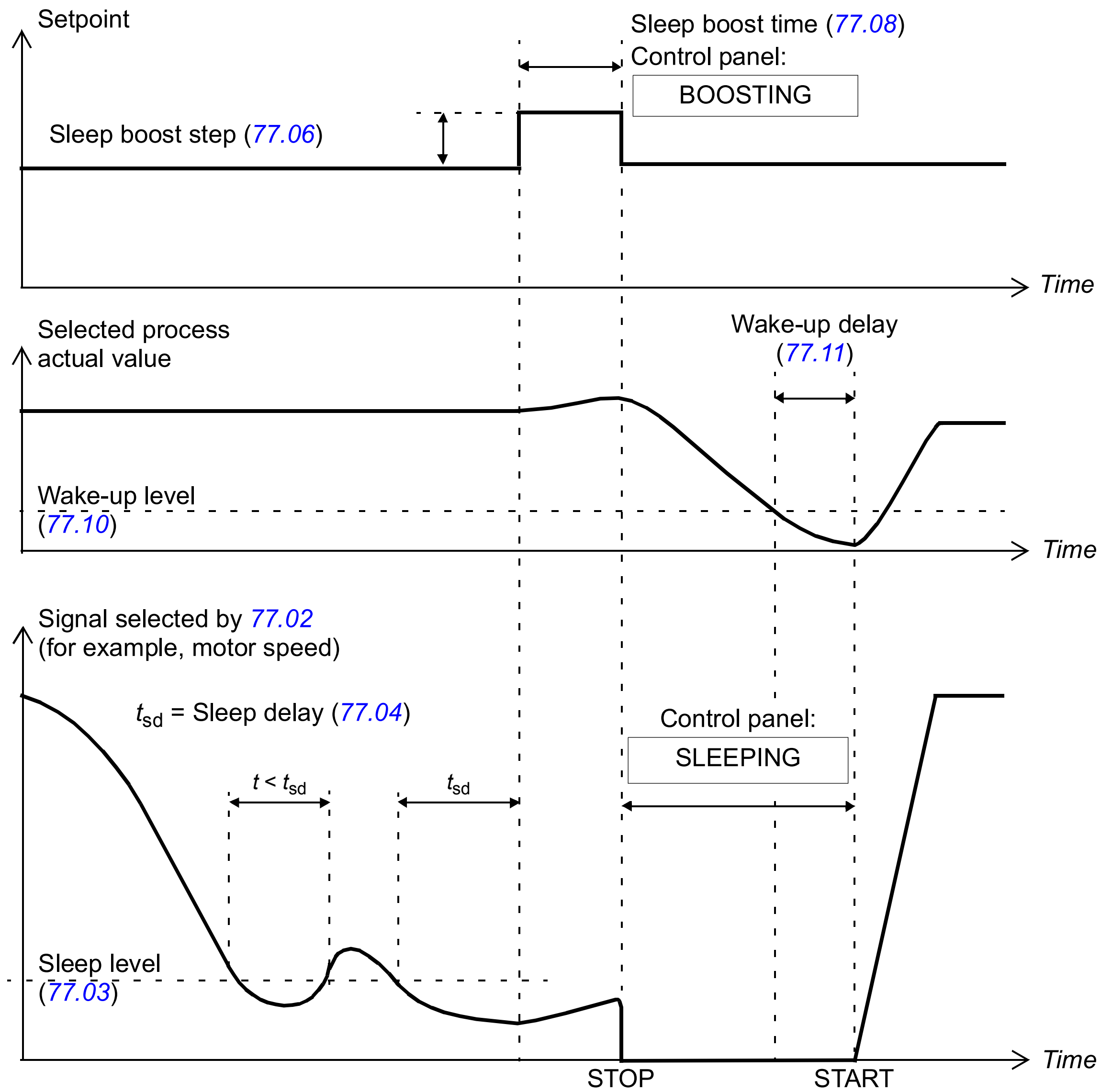
**Diagnostics**

Parameters *04.01*…*04.05* (page *123*), *04.20*…*04.25* (page *123*) and *06.20 Pump status word* (page *130*).

## Sleep function

The sleep function is suitable for PID control applications where the consumption varies, such as clean water pumping systems. When used, it stops the pump completely during low demand, instead of running the pump slowly below its efficient operating range. The following example visualizes the operation of the 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 *77 Pump sleep* (page *254*).

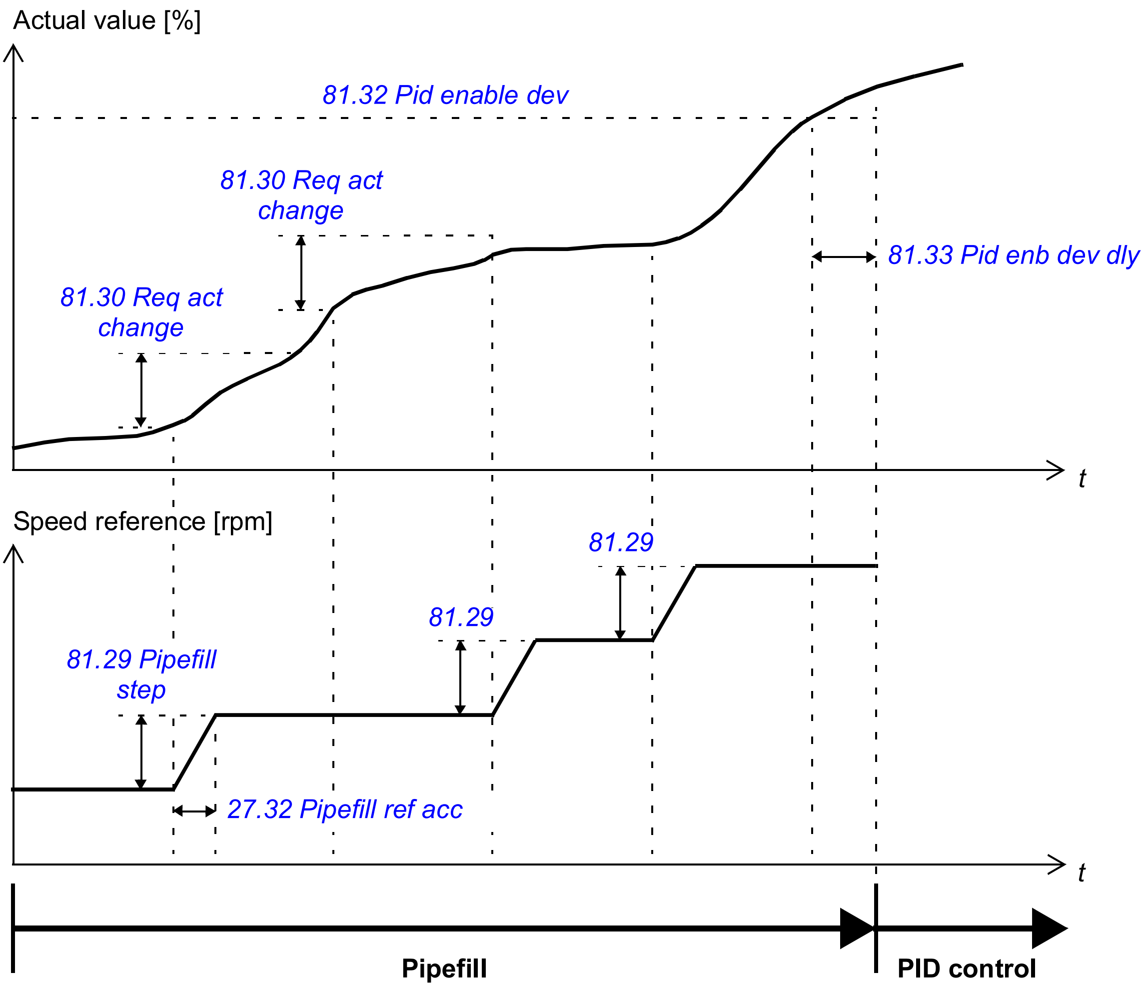
## Diagnostics

Parameters *04.26 Wake up level* (page *124*), *06.02 Status word2* (page *127*), *06.20 Pump status word* (page *130*) and *08.21 Pump alarm word* (page *134*).

## Soft pipefill

Filling up an empty system using normal PID control would cause a sudden pressure peak. To avoid this, a soft pipefill function is available. This involves running the pump at a lower speed until a predefined threshold of process actual value (for example, measured pressure) is achieved. If a specified increase in the actual value is not achieved within a specified time, pump speed is stepped up. This is repeated until the process actual value reaches the threshold level, after which normal process PID control is resumed. A timeout for the whole pipefill phase can also be defined.

The following drawing illustrates the operation of the pipefill function.




### Settings

Parameters *27.32 Pipefill ref acc* (page *190*) and *27.33 Pipefill ref dec* (page *190*); parameter group *81 Pump protection* (page *271*).

### Diagnostics

Parameters *06.20 Pump status word* (page *130*), *08.20 Pump fault word* (page *133*) and *08.21 Pump alarm word* (page *134*).

## Autochange

The Autochange function can be used to equalize duty time between multiple pumps by varying the sequence in which pumps are started as the required pumping capacity increases.

There are three autochange modes selectable by parameter *78.01 Autochg style*:

- Fixed intervals *(Fixed)*: The starting sequence is shifted periodically at pre-defined intervals (parameter *78.05 Autochg interval*). In traditional pump control, the pump speed must also be below the level defined by parameter *78.04 Autochg level*.
- Runtime equalization *(Hourcount)*: The starting sequence is rearranged when the difference between the runtimes of two pumps exceed a limit, *78.15 Runtime diff*. In the new sequence, the pump with the lowest runtime will be started first, the pump with the highest runtime will be started last.
- Autochange when stopped *(All stop)*: The starting sequence is shifted every time the drive (in traditional pump control) or the master drive (in multipump or level control) stops.

All pumps take part in the autochange sequence, except in a traditional pump control configuration where a fixed pump is controlled by the drive at all times and the remaining pumps are only turned on/off by the drive logic (an example is shown starting on page *96*). In this case, the fixed pump is always started first, and the starting sequence of the auxiliary pumps is determined by the autochange function.

#### Settings

Parameter group *78 Pump autochange* (page *257*).

#### Diagnostics

Parameters *04.29*…*04.36* (pump runtime counters; page *124*), *05.02 Trad pump cmd* (page *125*), *05.03 Trad master* (page *125*), *05.04 Nbr aux pumps on* (page *125*), *05.36 First in order* (page *126*), *05.37 Time autochg* (page *126*), *06.20 Pump status word* (page *130*), *08.21 Pump alarm word* (page *134*).

### Flow calculation

The flow calculation function provides a reasonably accurate (typically ±3…6%) calculation of the flow without the installation of a separate flow meter. The flow is calculated on the basis of parameter data such as pump inlet and outlet diameters, pressure at pump inlet and outlet, height difference of pressure sensors, and pump characteristics.

The user can either define a PQ (power/flow) or HQ (head/flow) performance curve that is used as the basis for the calculation. It is also possible to use both curve types together with a breakpoint setting.

**Notes:**

- The flow calculation function is not to be used for invoicing purposes.
- The flow calculation function cannot be used outside the normal operating range of the pump.

#### Settings

Parameter group *80 Flow calculation* (page *267*). The presence of pressure sensors in the system determines which parameters should be set; refer to the following table for recommendations.

| Parameter | Without pressure sensors | With pressure sensors |
|---|---|---|
| *80.01 Flow calc mode* | Typically *PQ curve* | Typically *HQ curve* |
| *80.02 Pump inlet sel* | Not required | Required |
| *80.03 Pump outlet sel* | Not required | Required |
| *80.04…80.13* (HQ curve definition) | Typically not required | Typically required |
| *80.14…80.23* (PQ curve definition) | Typically required | Typically not required |
| *80.25 Pump inlet diam* | Not required | Required |
| *80.26 Pump outlet diam* | Not required | Required |
| *80.27 Sensors hgt diff* | Not required | Required |
| *80.28 Pump nom speed* | Required | Required |
| *80.29 Density* | Not required | Required |
| *80.30 Efficiency* | Required | Not required |
| *80.31 Flow calc gain* | Optional correction factor | |
| *80.32 Calc low sp* | Optional pump speed low limit for calculation | |

#### Diagnostics

Parameters *05.05…05.08* (page *125*).

### Pump cleaning

The drive has a pump cleaning function that can be used to prevent solids from building up on pump impellers or piping. The function consists of a programmable sequence of forward and reverse runs of the pump to shake off any residue on the impeller or piping. This is especially useful with booster and wastewater pumps.

The cleaning sequence can be programmed to occur at suitable intervals, or whenever certain triggering conditions are met.

**Note:** Not all pumps can be rotated in the reverse direction.

#### Settings

Parameter group *82 Pump cleaning* (page *278*).

#### Diagnostics

Parameters *06.20 Pump status word* (page *130*), *08.20 Pump fault word* (page *133*) and *08.21 Pump alarm word* (page *134*).

## Protective functions

### Pressure monitoring

The control program contains protective functions for two-level analog or single-level digital pressure monitoring of both the inlet and outlet of the pump.

In analog monitoring, whenever the pressure being monitored meets the first limit, the drive indicates a warning, trips on a fault, or starts to follow a pre-defined reference. When the second limit is met, the drive either stops or trips on a fault.

In digital pressure monitoring, one limit is observed. Whenever the limit is met, the drive indicates an alarm, trips on a fault, or starts to follow a pre-defined reference.

### Flow monitoring

The control program has a monitoring function for flow that can be configured to generate an alarm or a fault whenever the flow falls below or rises above predefined limits.

The flow can either be calculated or measured using a flow meter connected to, for example, an analog input.

### Application profile monitoring

The application profile monitoring function can be used for long-term supervision of an actual signal. If the selected signal remains above the supervision limit for a specified time, an alarm is generated.

For example, monitoring the deviation between the PID controller setpoint and actual value (parameter *04.04 Process PID err*) gives an indication of the general condition of the pump, piping and valves. On the other hand, the PID controller output (parameter *04.05 Process PID out*) remaining at 100% for a long time would indicate a leak in the outlet piping.

### Settings

Parameter group *81 Pump protection* (page *271*).

### Diagnostics

Parameters *06.20 Pump status word* (page *130*), *08.20 Pump fault word* (page *133*) and *08.21 Pump alarm word* (page *134*).

# 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.

See also *Pressure sensor connection examples* on page *108*.

### Settings

Parameter group *13 Analogue inputs* (page *142*).

### Diagnostics

Parameters *02.04*…*02.13* (page *113*).

## Programmable analog outputs

The drive has two programmable current-type 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 *158*).

### Diagnostics

Parameters *02.16*…*02.19* (page *114*).

## Programmable digital inputs and outputs

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

One digital input (DI5) doubles as a PTC thermistor input. See section *Thermal motor protection* on page *78*.

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

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

### Settings

Parameter group *14 Digital I/O* (page *148*).

#### Diagnostics

Parameters *02.01 DI status* (page *113*), *02.03 DIO status* (page *113*), *02.20 Freq in* (page *114*) and *02.21 Freq out* (page *114*).

## 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 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:

| | Digital inputs (DI) | Digital I/O (DIO) | Analog inputs (AI) | Analog outputs (AO) | Relay outputs (RO) |
|---|---|---|---|---|---|
| JCU Control Unit | 6 | 2 | 2 | 2 | 2 |
| FIO-11 | - | 2 | 3 | 1 | - |
| FIO-21 | 1 | - | 1 | - | 2 |
| FIO-31 | - | - | - | - | 4 |

For example, with an FIO-21 connected to the drive, parameters controlling DI1…7, DIO1…4, AI1…3, AO1…2 and RO1…4 are in use.

#### Settings

Parameter groups *13 Analogue inputs* (page *142*), *14 Digital I/O* (page *148*), *15 Analogue outputs* (page *158*) and *94 Ext IO conf* (page *281*).

## Programmable relay outputs

The drive has two relay outputs. The signals 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 *148*).

#### Diagnostics

Parameter *02.02 RO status* (page *113*).

## 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 *329*) and *Control through a fieldbus adapter* (page *357*).

#### Settings

Parameter groups *50 Fieldbus* (page *229*), *51 FBA settings* (page *231*), *52 FBA data in* (page *232*), *53 FBA data out* (page *233*) and *58 Embedded Modbus* (page *234*).

# 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 *185*).

## 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 *184*).

## 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* 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:

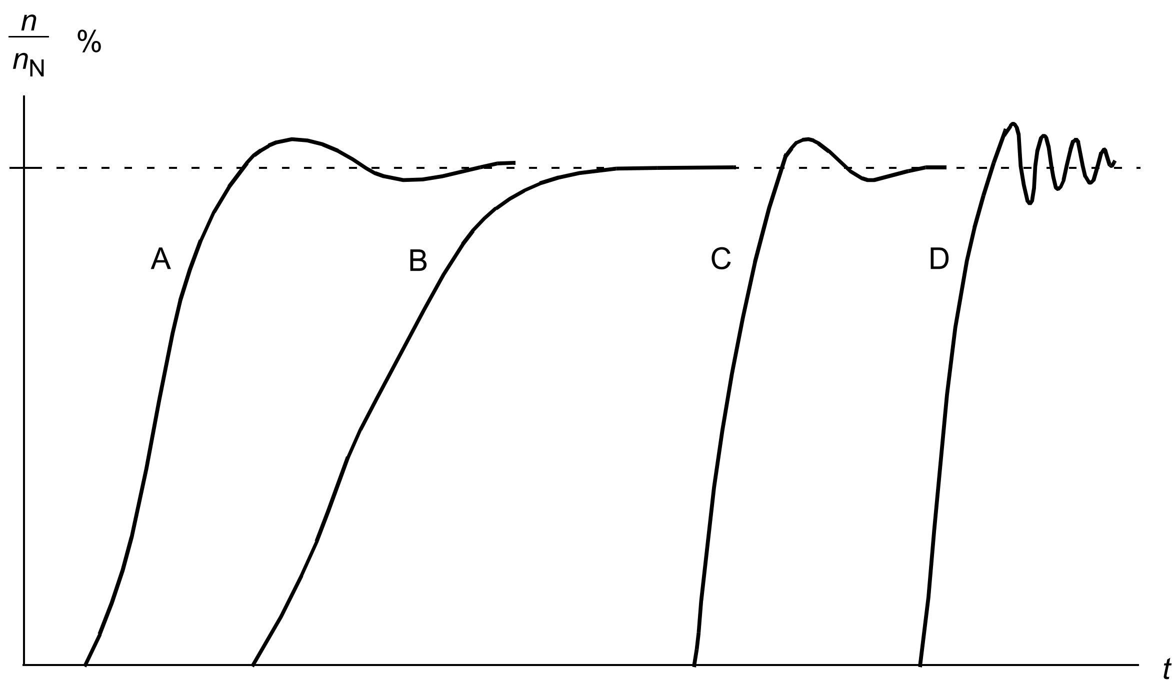
- The ID run has been successfully completed
- Speed, torque, current and acceleration limits (parameter groups *20 Limits* and *22 Speed ref ramp*) are set
- Speed feedback filtering, speed error filtering and zero speed are set (parameter groups *19 Speed calculation* and *23 Speed ctrl*)
- The drive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

The results of the autotune routine are automatically transferred into parameters

- *23.01 Proport gain* (proportional gain of the speed controller)

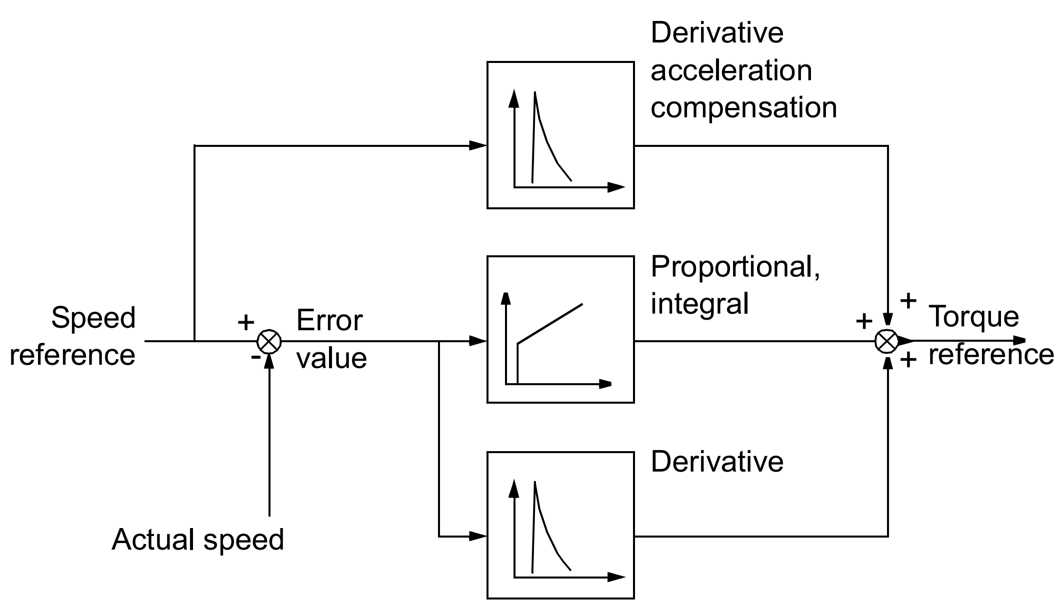
- *23.02 Integration time* (integration time of the speed controller)
- *01.31 Mech time const* (mechanical time constant of the machinery).

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 *176*).

#### Diagnostics

Parameters *01.01 Motor speed rpm* (page *112*), *01.02 Motor speed %* (page *112*) and *01.14 Motor speed est* (page *112*).

### 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).

In scalar control, some standard features are not available.

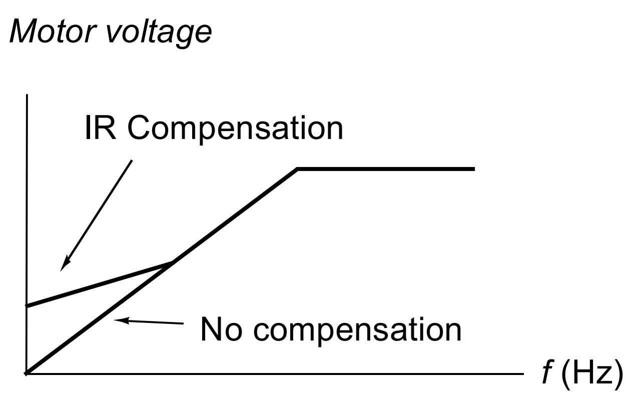
#### Settings

Parameter *99.05 Motor ctrl mode* (page *283*).

#### 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.



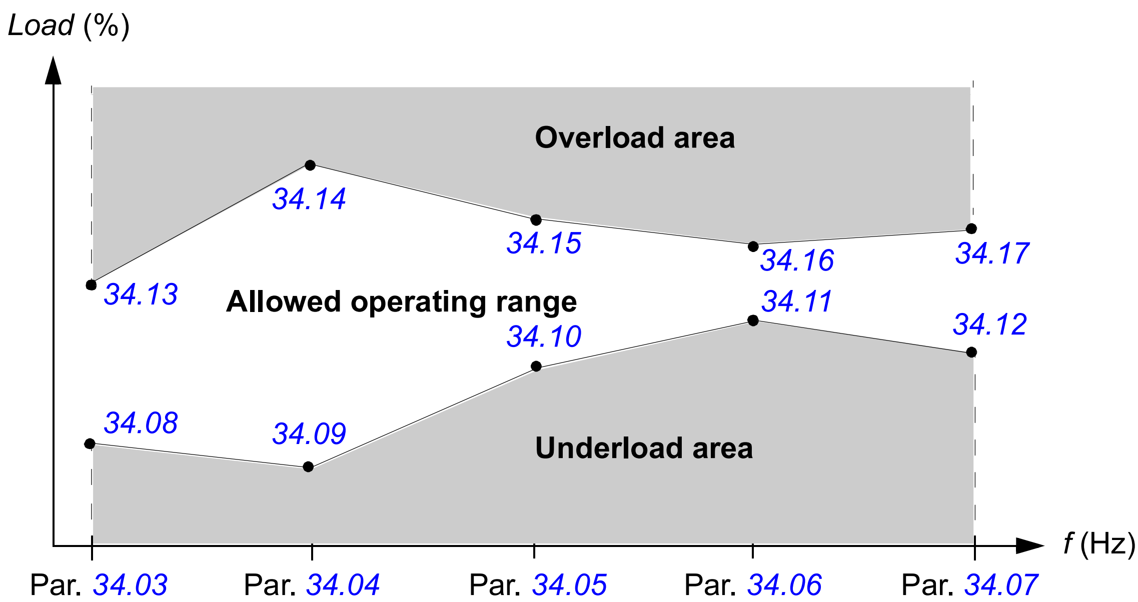

#### Settings

Parameter *40.07 IR-compensation* (page *220*).

## User-definable load curve

The drive output can be limited by defining a 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.

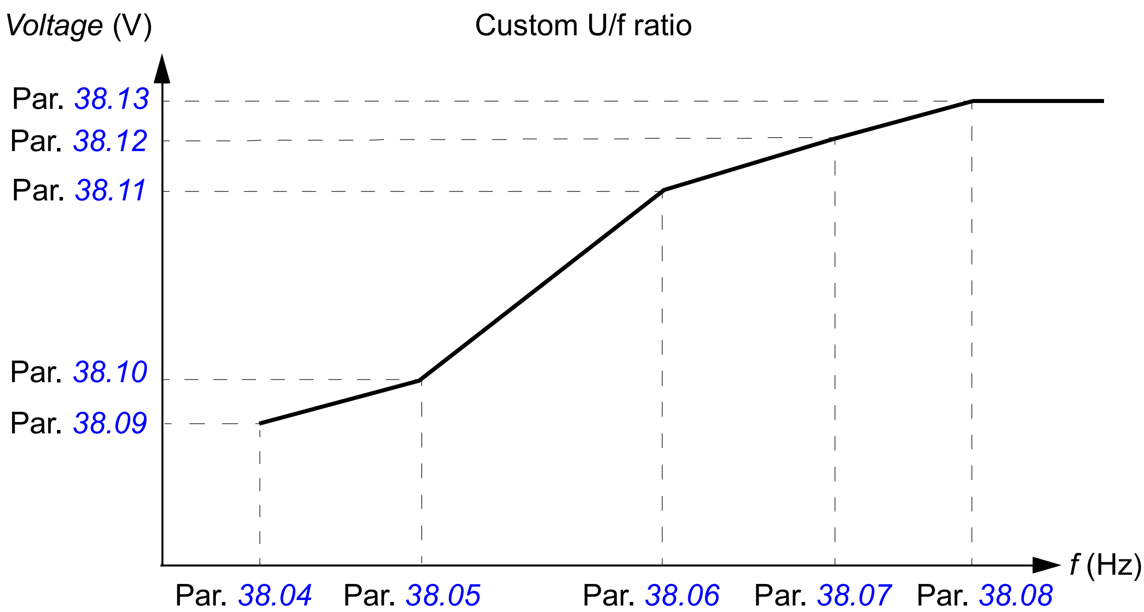



### Settings

Parameter group *34 User load curve* (page *205*).

## 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).




**Notes:**

- The *U/f* curve can be used in scalar control only, i.e., when parameter *99.05 Motor ctrl mode* is set to *Scalar*.
- Each user-defined point defined must have a higher frequency and higher voltage than the previous point.

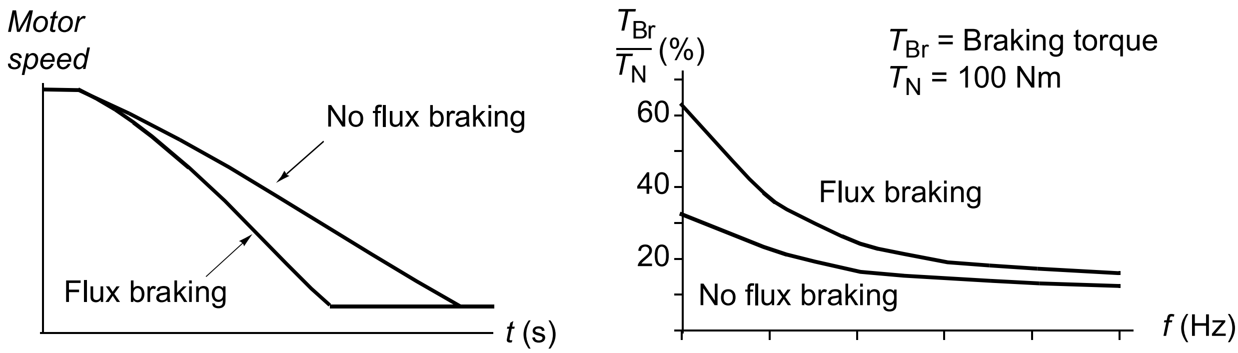
**WARNING!** High voltage at low frequencies may result in poor performance or motor damage due to overheating.

**Settings**

Parameter group *38 Flux ref* (page *218*).

## 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.

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 *220*).

# Application control

## Application macros

See chapter *Application macros* (page *87*).

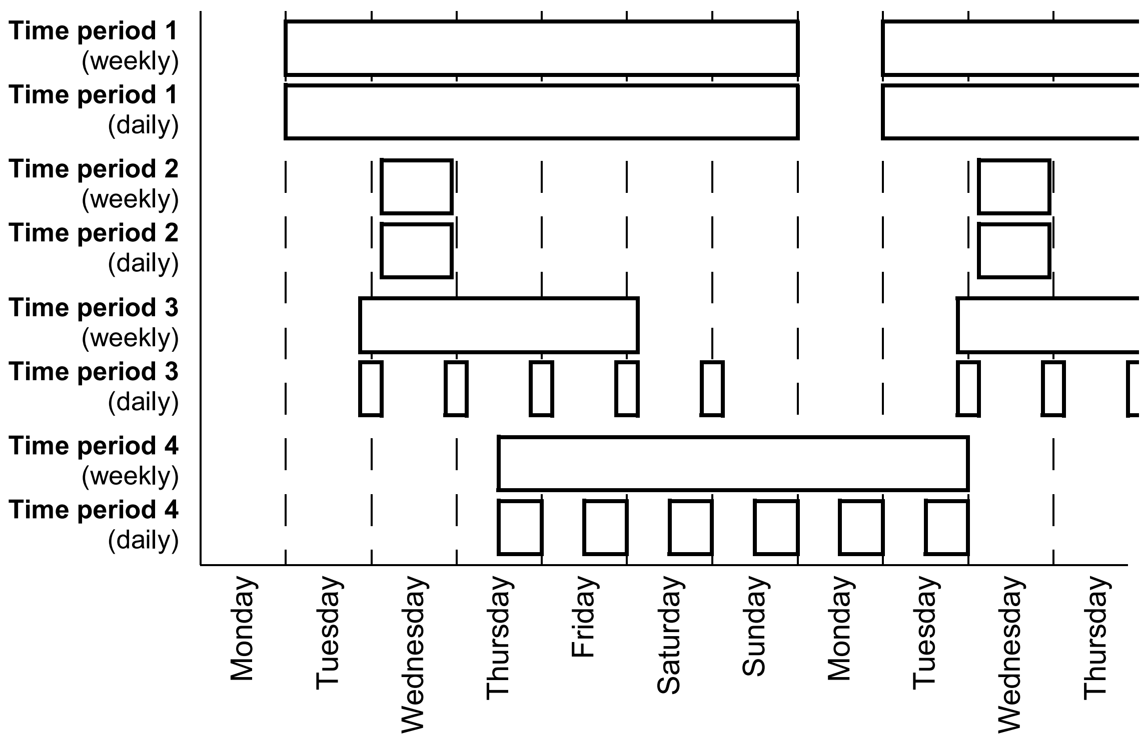
## Timers

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 *109*). In addition, bit 4 of parameter *06.14* is on if any one of the four timers is on.

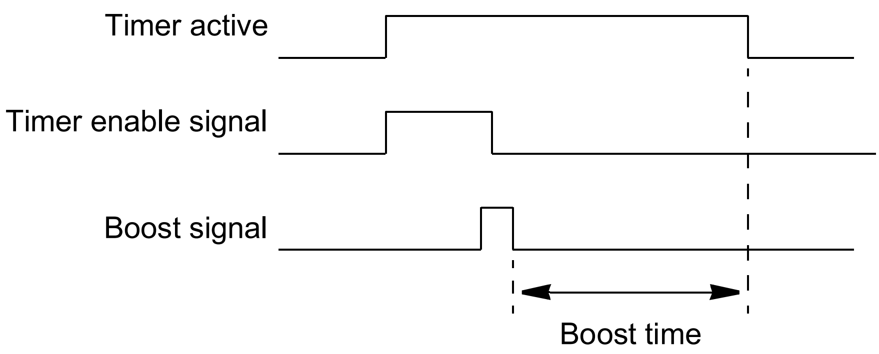
Each time period can be assigned to multiple timers; likewise, a timer can be controlled by multiple time periods.

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 *214*).

### Diagnostics

Parameter *06.14 Timed func stat* (page *129*).

# DC voltage control

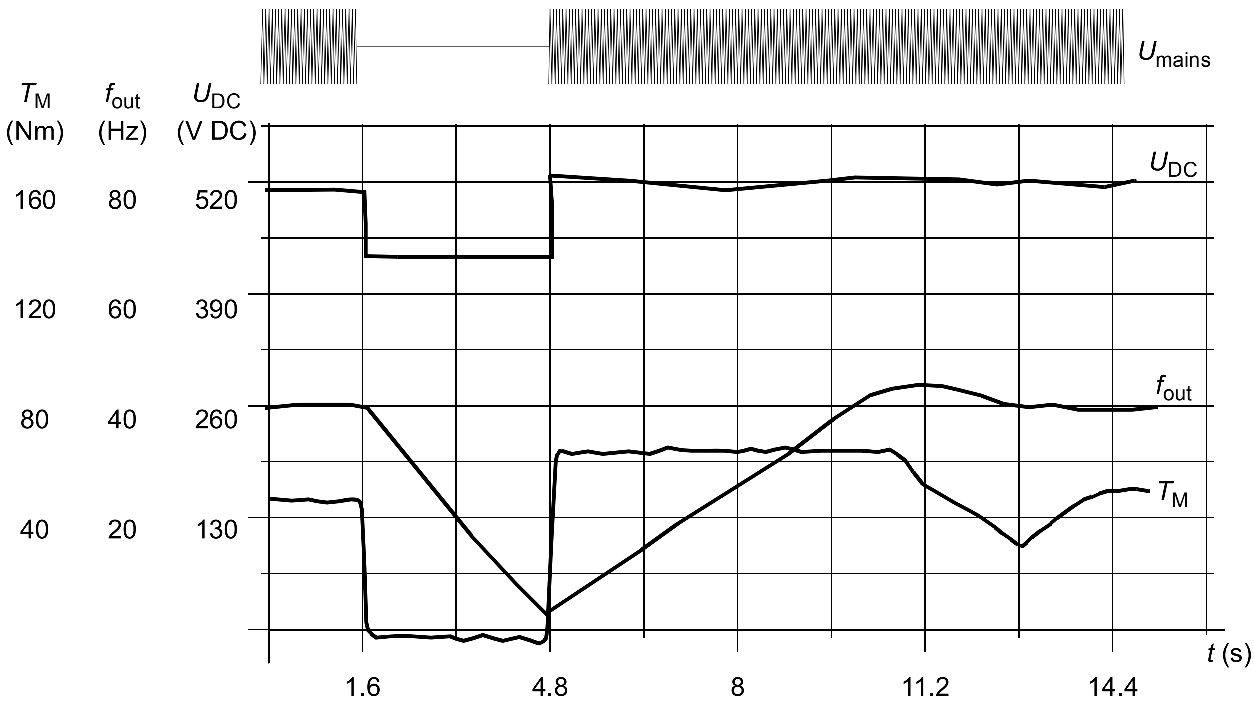
## Overvoltage control

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.

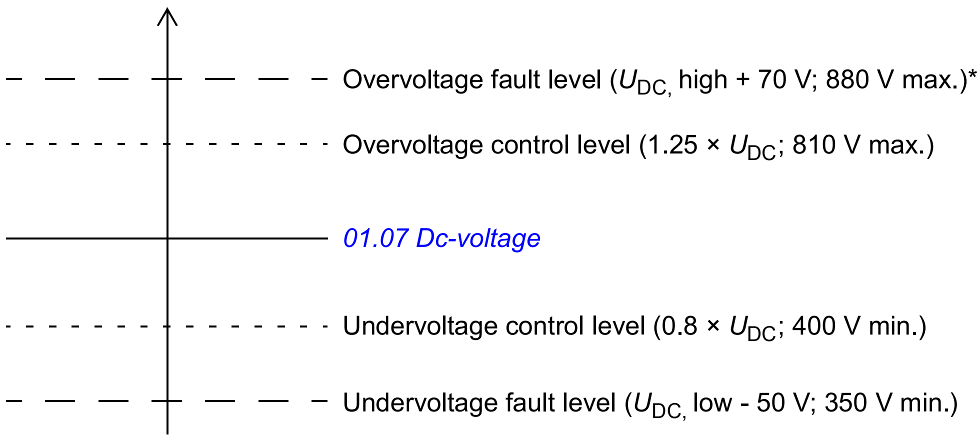



$U_{DC}$= intermediate circuit voltage of the drive, $f_{out}$ = output frequency of the drive,
$T_M$ = motor torque
Loss of supply voltage at nominal load ($f_{out}$ = 40 Hz). The intermediate circuit DC voltage drops to the minimum limit. The controller keeps the voltage steady as long as the mains 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 SupplyVoltAutoId*; the user can then define the voltage manually at parameter *47.04 Supply voltage*.




$U_{DC}$ = 1.35 × *01.19 Used supply volt*

$U_{DC}$, high = 1.25 x $U_{DC}$

$U_{DC}$, low = 0.8 x $U_{DC}$

*Drives with 230 V supply voltage (ACQ810-04-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 is stabilized.

### Settings

Parameter group *47 Voltage ctrl* (page *228*).

### Diagnostics

Parameters *01.07 Dc-voltage* (page *112*), *01.19 Used supply volt* (page *112*) and *06.05 Limit word1* (page *128*).

# 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 1…3 PTC 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.

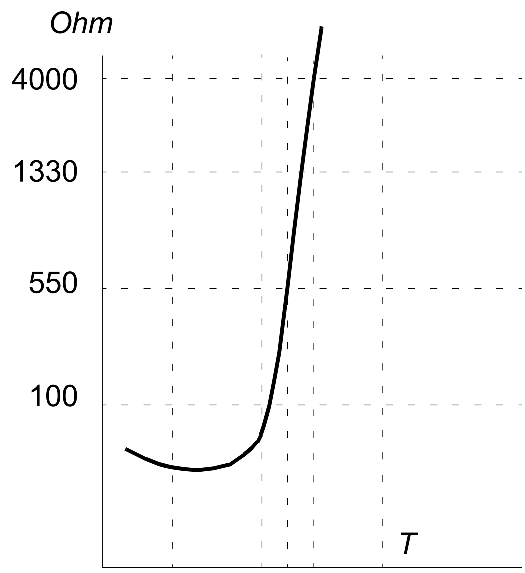
### Temperature measurement

It is possible to detect motor overtemperature by connecting a motor temperature sensor between +24 V and digital input DI5 on the drive.

Constant current is fed through the sensor. 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 below shows typical PTC sensor resistance values as a function of the motor operating temperature.

| Temperature | PTC resistance |
|---|---|
| Normal | 0…1 kohm |
| Excessive | ≥ 1 kohm |




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

For further information on the wiring, refer to the *Hardware Manual* of the drive.

#### Settings

Parameter group *31 Motor therm prot* (page *197*).

#### Diagnostics

Parameters *01.17 Motor temp1* (page *112*) and *01.18 Motor temp2* (page *112*).

## 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 reset

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 *202*).

#### Diagnostics

Parameter *08.07 Alarm word3* (page *132*).

# Diagnostics

## Energy savings 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 $CO_2$ emission, and
- A load analyzer showing the load profile of the drive (see section *Load analyzer* on page *82*).

**Note:** The accuracy of the energy savings calculation is directly dependent on the accuracy of the reference motor power given in parameter *45.08 Pump ref power*.

### Settings

Parameter group *45 Energy optimising* (page *227*).

## Energy consumption monitoring

The control program monitors the energy consumption of the drive and pump, and provides the consumption during the last 12 calendar months as actual signals.

There is also a comparison function that generates an alarm if the consumption rises significantly compared to past consumption. The length of a monitoring period is set by a parameter. The energy consumption within the currently running period is compared to a parameter-adjustable limit, the previous monitoring period, or the average of two previous periods. Whenever the consumption within the current period exceeds the reference by a predefined margin (or tolerance), an alarm is given.

### Settings

Parameter group *83 Energy monitoring* (page *280*).

### Diagnostics

Parameters *05.20*…*05.35* (page *125*).

## 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 *202*).

#### Diagnostics

Parameter *06.13 Superv status* (page *129*).

### 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:

- Ontime 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*).

#### Diagnostics

Parameters *04.09*…*04.14* (page *123*) and *06.15 Counter status* (page *129*).

### Load analyzer

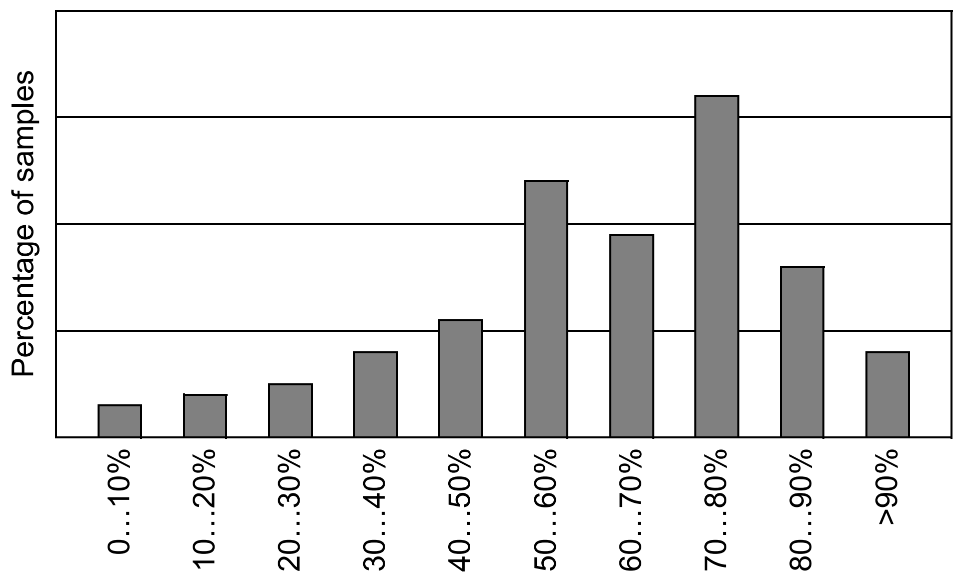
#### Peak value logger

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 ranges
(parameters *64.24*…*64.33*)

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 and diagnostics

Parameter group *64 Load analyzer* (page *238*).

# 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 *164*).

### 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 *228*).

## Drive-to-drive link

The drive-to-drive (D2D) link is a daisy-chained RS-485 transmission line that allows basic master/follower communication with one master drive and multiple followers.

The drive-to-drive link is used for connecting drives when forming a station with multiple pumps.

### Settings

Parameter group *76 MF communication* (page *250*).

5

# Application macros

## What this chapter contains

This chapter describes the intended use, operation, default control connections, start-up procedure and an application example of each application macro.

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. User parameter sets are managed by the parameters in group *16 System* (page *164*).

Application macros are activated through the control panel Main menu by selecting ASSISTANTS – Application Macro. A few basic questions about the application appear on the panel; based on the answers, the most suitable macro is applied by the drive. Parameter *16.20 Macro selected* indicates which application macro is active.

After the activation of an application macro, an assistant can optionally be launched to set up the essential configuration parameters related to the application. Each of these assistants can also be invoked later by selecting ASSISTANTS in the control panel Main menu.

# Factory default macro

## Description and typical application

This macro is suitable for a pump station consisting of one drive controlling a single pump. The system can consist of e.g. one ACQ810 drive, one pump, and a sensor. The sensor typically measures either flow or pressure, and is located at the output of the pump.

By default, process reference (setpoint) is set to 40%, but can alternatively be adjusted through e.g. analog input AI1. The process actual value, or feedback signal, should be connected to analog input AI2. The start command is given through digital input DI1.

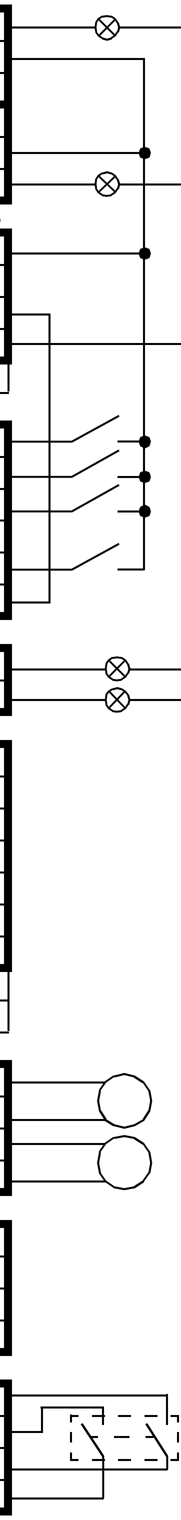
The sleep function is activated to optimize the energy efficiency of the installation. By default, the drive is stopped if the motor speed is below 20% of its nominal speed for longer than 60 seconds.

## Default settings

See chapter *Additional parameter data* (page *287*).

## Default control connections

| | | |
|---|---|---|
| | | **XPOW** |
| External power input<br>24 V DC, 1.6 A | +24VI | 1 |
| | GND | 2 |
| | | **XRO1, XRO2** |
| Relay output RO1 [Ready]<br>250 V AC / 30 V DC<br>2 A | NO | 1 |
| | COM | 2 |
| | NC | 3 |
| Relay output RO2 [Fault(-1)]<br>250 V AC / 30 V DC<br>2 A | NO | 4 |
| | COM | 5 |
| | NC | 6 |
| | | **XD24** |
| +24 V DC | +24VD | 1 |
| Digital input ground | DIGND | 2 |
| +24 V DC | +24VD | 3 |
| Digital input/output ground | DIOGND | 4 |
| Ground selection jumper | | |
| | | **XDI** |
| Digital input DI1 [Stop/Start] | DI1 | 1 |
| Digital input DI2 [Constant speed 1] | DI2 | 2 |
| Digital input DI3 [Reset] | DI3 | 3 |
| Digital input DI4 | DI4 | 4 |
| Digital input DI5 [EXT1/EXT2 selection] | DI5 | 5 |
| Start interlock (0 = Stop) | DIIL | A |
| | | **XDIO** |
| Digital input/output DIO1 [Output: Ready] | DIO1 | 1 |
| Digital input/output DIO2 [Output: Running] | DIO2 | 2 |
| | | **XAI** |
| Reference voltage (+) | +VREF | 1 |
| Reference voltage (–) | -VREF | 2 |
| Ground | AGND | 3 |
| Analog input AI1 (Current or voltage, selectable by jumper AI1)<br>[Current] [Speed reference 1] | AI1+ | 4 |
| | AI1- | 5 |
| Analog input AI2 (Current or voltage, selectable by jumper AI2)<br>[Current] [Process actual value 1*] | AI2+ | 6 |
| | AI2- | 7 |
| AI1 current/voltage selection jumper | | AI1 |
| AI2 current/voltage selection jumper | | AI2 |
| | | **XAO** |
| Analog output AO1 [Current] | AO1+ | 1 |
| | AO1- | 2 |
| Analog output AO2 [Speed rpm] | AO2+ | 3 |
| | AO2- | 4 |
| | | **XD2D** |
| Drive-to-drive link termination jumper | | T |
| Drive-to-drive link. | B | 1 |
| | A | 2 |
| | BGND | 3 |
| | | **XSTO** |
| Safe torque off. Both circuits must be closed for the drive to start. | OUT1 | 1 |
| | OUT2 | 2 |
| | IN1 | 3 |
| | IN2 | 4 |
| Control panel connection | | |
| Memory unit connection | | |



*See *Pressure sensor connection examples* on page *108*.

# External control macro

## Description and typical application

The external control macro can be used in single-pump systems where the process is controlled by other devices than the drive. The drive is speed-controlled.

By default, the drive receives a speed reference through analog input AI1. The reference can alternatively be received through one of the supported fieldbus adapters.

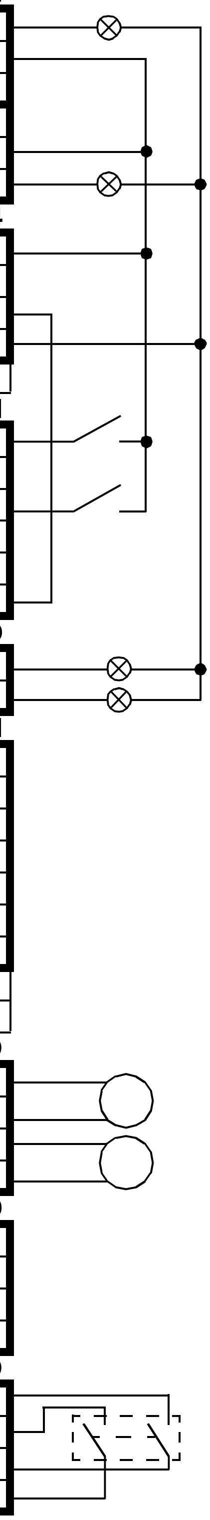
## Default settings

Below is a listing of default parameter values that differ from those listed in chapter *Additional parameter data* (page *287*).

| Parameter | | External control macro default |
|---|---|---|
| **No.** | **Name** | |
| *12.01* | *Ext1/Ext2 sel* | *Ext1* |
| *16.16* | *Menu set active* | *Ext short* |
| *16.20* | *Macro selected* | *Ext ctrl* |
| *26.02* | *Const speed sel1* | C.FALSE |
| *77.01* | *Sleep mode sel* | *Not used* |

## Default control connections

| | | |
|---|---|---|
| | | **XPOW** |
| External power input 24 V DC, 1.6 A | +24VI | 1 |
| | GND | 2 |
| | | **XRO1, XRO2** |
| Relay output RO1 [Ready] 250 V AC / 30 V DC 2 A | NO | 1 |
| | COM | 2 |
| | NC | 3 |
| Relay output RO2 [Fault(-1)] 250 V AC / 30 V DC 2 A | NO | 4 |
| | COM | 5 |
| | NC | 6 |
| | | **XD24** |
| +24 V DC | +24VD | 1 |
| Digital input ground | DIGND | 2 |
| +24 V DC | +24VD | 3 |
| Digital input/output ground | DIOGND | 4 |
| Ground selection jumper | | |
| | | **XDI** |
| Digital input DI1 [Stop/Start] | DI1 | 1 |
| Digital input DI2 | DI2 | 2 |
| Digital input DI3 [Reset] | DI3 | 3 |
| Digital input DI4 | DI4 | 4 |
| Digital input DI5 | DI5 | 5 |
| Start interlock (0 = Stop) | DIIL | A |
| | | **XDIO** |
| Digital input/output DIO1 [Output: Ready] | DIO1 | 1 |
| Digital input/output DIO2 [Output: Running] | DIO2 | 2 |
| | | **XAI** |
| Reference voltage (+) | +VREF | 1 |
| Reference voltage (–) | -VREF | 2 |
| Ground | AGND | 3 |
| Analog input AI1 (Current or voltage, selectable by jumper AI1) [Current] [Speed reference 1] | AI1+ | 4 |
| | AI1- | 5 |
| Analog input AI2 (Current or voltage, selectable by jumper AI2) [Current] [Process actual value 1*] | AI2+ | 6 |
| | AI2- | 7 |
| AI1 current/voltage selection jumper | | AI1 |
| AI2 current/voltage selection jumper | | AI2 |
| | | **XAO** |
| Analog output AO1 [Current] | AO1+ | 1 |
| | AO1- | 2 |
| Analog output AO2 [Speed rpm] | AO2+ | 3 |
| | AO2- | 4 |
| | | **XD2D** |
| Drive-to-drive link termination jumper | | T |
| Drive-to-drive link. | B | 1 |
| | A | 2 |
| | BGND | 3 |
| | | **XSTO** |
| Safe torque off. Both circuits must be closed for the drive to start. | OUT1 | 1 |
| | OUT2 | 2 |
| | IN1 | 3 |
| | IN2 | 4 |
| Control panel connection | | |
| Memory unit connection | | |



*See *Pressure sensor connection examples* on page *108*.

# Hand/Auto control macro

## Description and typical application

Start and stop commands and references (setpoints) can be given from one of two external control locations, EXT1 (Hand) or EXT2 (Auto). The start/stop commands received through EXT1 (Hand) are connected to digital input DI1, while the reference is connected to analog input AI1. The start/stop commands from EXT2 (Auto) are connected to DI2 while the reference is connected to AI2. The selection between Hand/Auto is dependent on the status of DI5. The drive is speed-controlled. The speed reference and start/stop commands can also be given from the control panel.

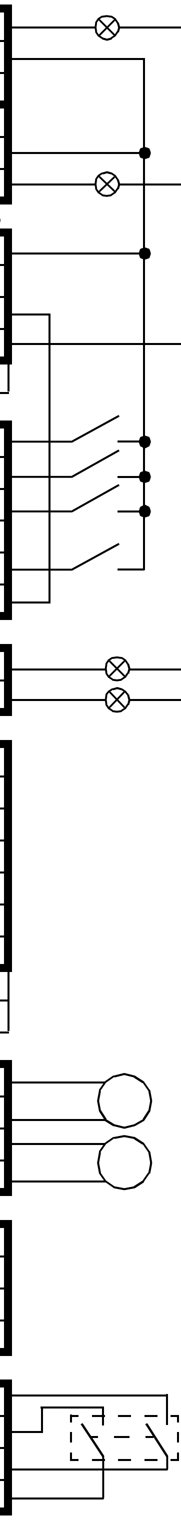
## Default settings

Below is a listing of default parameter values that differ from those listed in chapter *Additional parameter data* (page 287).

| Parameter | | Hand/Auto macro default |
|---|---|---|
| **No.** | **Name** | |
| *10.05* | *Ext2 start in1* | *DI2* |
| *12.05* | *Ext2 ctrl mode* | *Speed* |
| *16.16* | *Menu set active* | *H/A short* |
| *16.20* | *Macro selected* | *Hand/Auto* |
| *21.02* | *Speed ref2 sel* | *AI2 scaled* |
| *26.02* | *Const speed sel1* | C.FALSE |
| *77.01* | *Sleep mode sel* | *Not used* |

## Default control connections

| | | |
|---|---|---|
| | | **XPOW** |
| External power input<br>24 V DC, 1.6 A | +24VI | 1 |
| | GND | 2 |
| | | **XRO1, XRO2** |
| Relay output RO1 [Ready]<br>250 V AC / 30 V DC<br>2 A | NO | 1 |
| | COM | 2 |
| | NC | 3 |
| Relay output RO2 [Fault(-1)]<br>250 V AC / 30 V DC<br>2 A | NO | 4 |
| | COM | 5 |
| | NC | 6 |
| | | **XD24** |
| +24 V DC | +24VD | 1 |
| Digital input ground | DIGND | 2 |
| +24 V DC | +24VD | 3 |
| Digital input/output ground | DIOGND | 4 |
| Ground selection jumper | | |
| | | **XDI** |
| Digital input DI1 [Stop/Start, Hand] | DI1 | 1 |
| Digital input DI2 [Stop/Start, Auto] | DI2 | 2 |
| Digital input DI3 [Reset] | DI3 | 3 |
| Digital input DI4 | DI4 | 4 |
| Digital input DI5 [Hand/Auto selection] | DI5 | 5 |
| Start interlock (0 = Stop) | DIIL | A |
| | | **XDIO** |
| Digital input/output DIO1 [Output: Ready] | DIO1 | 1 |
| Digital input/output DIO2 [Output: Running] | DIO2 | 2 |
| | | **XAI** |
| Reference voltage (+) | +VREF | 1 |
| Reference voltage (–) | -VREF | 2 |
| Ground | AGND | 3 |
| Analog input AI1 (Current or voltage, selectable by jumper AI1)<br>[Current] [Speed reference 1, Hand] | AI1+ | 4 |
| | AI1- | 5 |
| Analog input AI2 (Current or voltage, selectable by jumper AI2)<br>[Current] [Speed reference 2, Auto] | AI2+ | 6 |
| | AI2- | 7 |
| AI1 current/voltage selection jumper | | AI1 |
| AI2 current/voltage selection jumper | | AI2 |
| | | **XAO** |
| Analog output AO1 [Current] | AO1+ | 1 |
| | AO1- | 2 |
| Analog output AO2 [Speed rpm] | AO2+ | 3 |
| | AO2- | 4 |
| | | **XD2D** |
| Drive-to-drive link termination jumper | | T |
| Drive-to-drive link. | B | 1 |
| | A | 2 |
| | BGND | 3 |
| | | **XSTO** |
| Safe torque off. Both circuits must be closed for the drive to start. | OUT1 | 1 |
| | OUT2 | 2 |
| | IN1 | 3 |
| | IN2 | 4 |
| Control panel connection | | |
| Memory unit connection | | |



# TRAD (Traditional pump) control macro

## Description and typical application

This macro is suitable for a pump station where one pump at a time is directly controlled by the drive, and the rest of the pumps are direct-on-line and switched on and off by the drive via a relay/contactor system. It is possible to have one pump permanently connected to the drive, or to connect any one of the pumps to the drive using contactors. The drive is capable of controlling up to eight parallel pumps.

By default, process reference (setpoint) is set to 40%, but can alternatively be adjusted through e.g. analog input AI1. The process actual value, or feedback signal, should be connected to analog input AI2. The start command is given through digital input DI1. Relay outputs are used to control auxiliary pumps.

## Default settings

Below is a listing of default parameter values that differ from those listed in chapter *Additional parameter data* (page *287*).

| Parameter | | Trad. pump control macro default |
|---|---|---|
| No. | Name | |
| *14.07* | *DIO2 out src* | *Fault(-1)* |
| *14.42* | *RO1 src* | *Trad pump1* |
| *14.45* | *RO2 src* | *Trad pump2* |
| *16.16* | *Menu set active* | *Trad short* |
| *16.20* | *Macro selected* | *Trad ctrl* |
| *26.02* | *Const speed sel1* | C.FALSE |
| *75.01* | *Operation mode* | *Trad ctrl* |
| *75.02* | *Nbr of pumps* | 2 |
| *75.25* | *Drive start dly* | 1 s |
| *78.01* | *Autochg style* | *All stop* |
| *78.03* | *Interlock mode* | *On* |
| *78.06* | *Interlock pump 1* | *DI2* |
| *78.07* | *Interlock pump 2* | *DI4* |

## Default control connections

| | | |
|---|---|---|
| | | **XPOW** |
| External power input 24 V DC, 1.6 A | +24VI | 1 |
| | GND | 2 |
| | | **XRO1, XRO2** |
| Relay output RO1 [Start pump 1] 250 V AC / 30 V DC 2 A | NO | 1 |
| | COM | 2 |
| | NC | 3 |
| Relay output RO2 [Start pump 2] 250 V AC / 30 V DC 2 A | NO | 4 |
| | COM | 5 |
| | NC | 6 |
| | | **XD24** |
| +24 V DC | +24VD | 1 |
| Digital input ground | DIGND | 2 |
| +24 V DC | +24VD | 3 |
| Digital input/output ground | DIOGND | 4 |
| Ground selection jumper | | |
| | | **XDI** |
| Digital input DI1 [Stop/Start] | DI1 | 1 |
| Digital input DI2 [Interlock pump 1] | DI2 | 2 |
| Digital input DI3 [Reset] | DI3 | 3 |
| Digital input DI4 [Interlock pump 2] | DI4 | 4 |
| Digital input DI5 [EXT1/EXT2 selection] | DI5 | 5 |
| Start interlock (0 = Stop) | DIIL | A |
| | | **XDIO** |
| Digital input/output DIO1 [Output: Ready] | DIO1 | 1 |
| Digital input/output DIO2 [Output: Fault(-1)] | DIO2 | 2 |
| | | **XAI** |
| Reference voltage (+) | +VREF | 1 |
| Reference voltage (–) | -VREF | 2 |
| Ground | AGND | 3 |
| Analog input AI1 (Current or voltage, selectable by jumper AI1) [Current] [Speed reference 1] | AI1+ | 4 |
| | AI1- | 5 |
| Analog input AI2 (Current or voltage, selectable by jumper AI2) [Current] [Process actual value 1*] | AI2+ | 6 |
| | AI2- | 7 |
| AI1 current/voltage selection jumper | | AI1 |
| AI2 current/voltage selection jumper | | AI2 |
| | | **XAO** |
| Analog output AO1 [Current] | AO1+ | 1 |
| | AO1- | 2 |
| Analog output AO2 [Speed rpm] | AO2+ | 3 |
| | AO2- | 4 |
| | | **XD2D** |
| Drive-to-drive link termination jumper | | T |
| Drive-to-drive link. | B | 1 |
| | A | 2 |
| | BGND | 3 |
| | | **XSTO** |
| Safe torque off. Both circuits must be closed for the drive to start. | OUT1 | 1 |
| | OUT2 | 2 |
| | IN1 | 3 |
| | IN2 | 4 |
| Control panel connection | | |
| Memory unit connection | | |

*See *Pressure sensor connection examples* on page *108*.

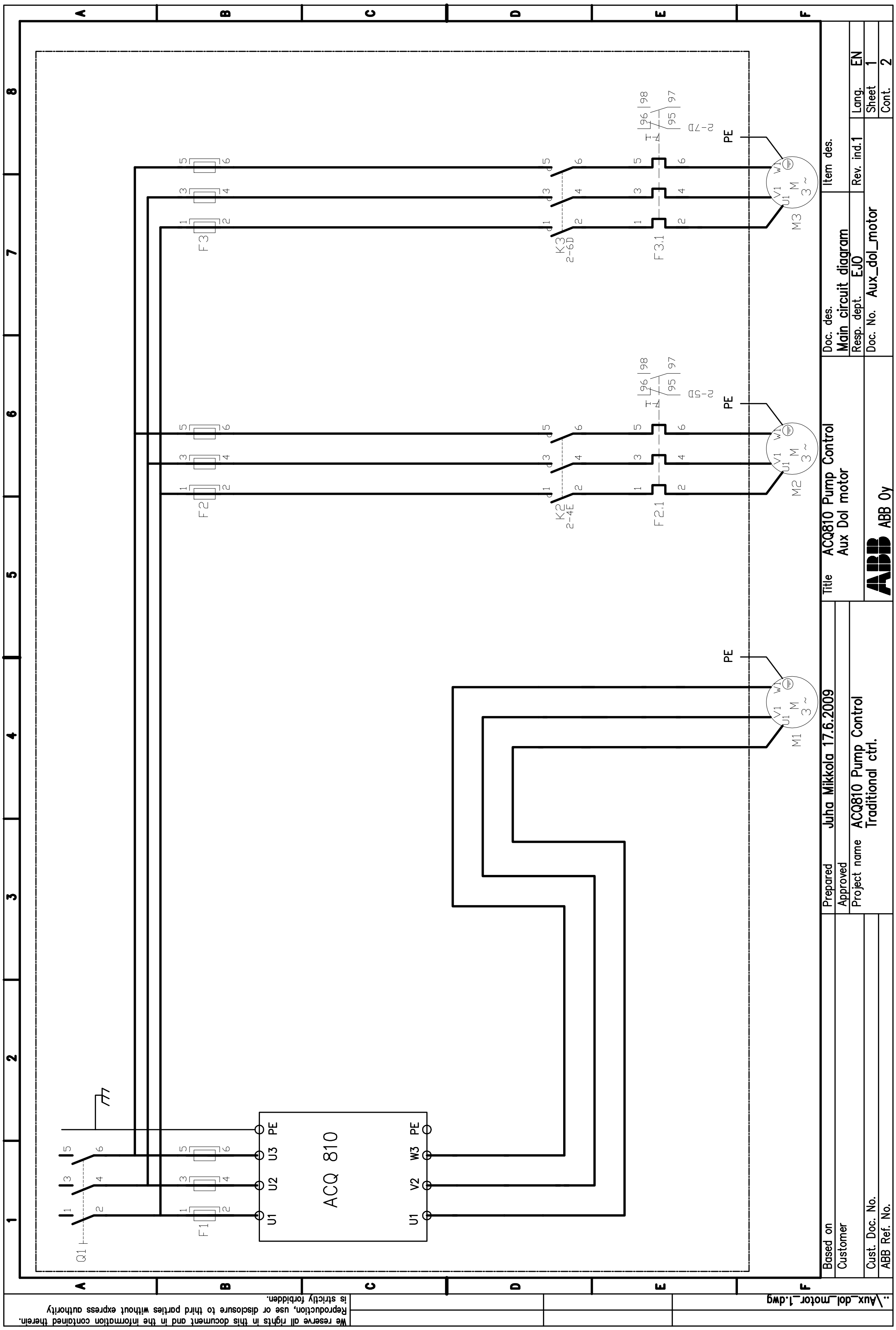
## Application examples

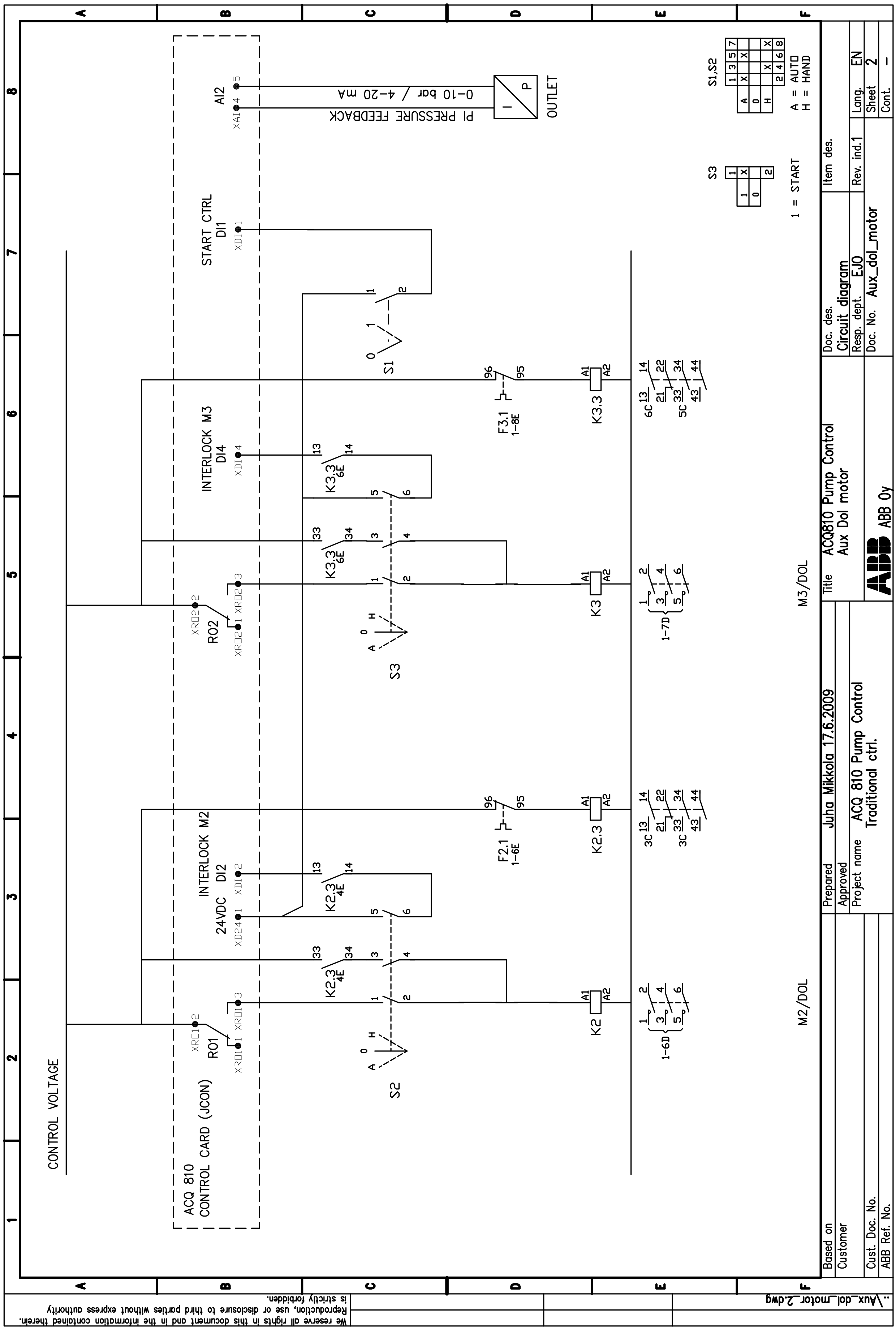
### Fixed drive-controlled pump with direct-on-line auxiliary pumps

In this example, the drive always controls the same pump. Auxiliary pumps are connected to the supply through contactors that are controlled by the drive.

Below is a listing of typical parameter values that would be used in this configuration.

| **Parameter** | | **Trad. pump control macro default** |
|---|---|---|
| **No.** | **Name** | |
| *14.42* | *RO1 src* | *Ready* |
| *14.45* | *RO2 src* | *Trad pump2* |
| *14.48* | *RO3 src* | *Trad pump3* |
| *16.20* | *Macro selected* | *Trad ctrl* |
| *26.02* | *Const speed sel1* | C.FALSE |
| *75.01* | *Operation mode* | *Trad ctrl* |
| *75.02* | *Nbr of pumps* | 3 |
| *75.25* | *Drive start dly* | 1 s |
| *78.01* | *Autochg style* | *All stop* |
| *78.02* | *Autochg trad* | *Aux* |
| *78.03* | *Interlock mode* | *On* |
| *78.06* | *Interlock pump 1* | *DI2* |
| *78.07* | *Interlock pump 2* | *DI4* |
| *78.07* | *Interlock pump 3* | *Not used* |


ACQ 810
U1
U2
U3
PE
U1
V2
W3
PE
Q1
F1
F2
F3
K2
2-4E
K3
2-6D
F2.1
F3.1
2-5D
2-7D
M1
M2
M3
M
3~
PE
Based on
Customer
Cust. Doc. No.
ABB Ref. No.
Prepared Juha Mikkola 17.6.2009
Approved
Project name ACQ810 Pump Control
Traditional ctrl.
Title ACQ810 Pump Control
Aux Dol motor
ABB Oy
Doc. des.
Main circuit diagram
Resp. dept. EJO
Doc. No. Aux_dol_motor
Item des.
Rev. ind.1
Lang. EN
Sheet 1
Cont. 2
We reserve all rights in this document and in the information contained therein. Reproduction, use or disclosure to third parties without express authority is strictly forbidden.
:\Aux_dol_motor_1.dwg

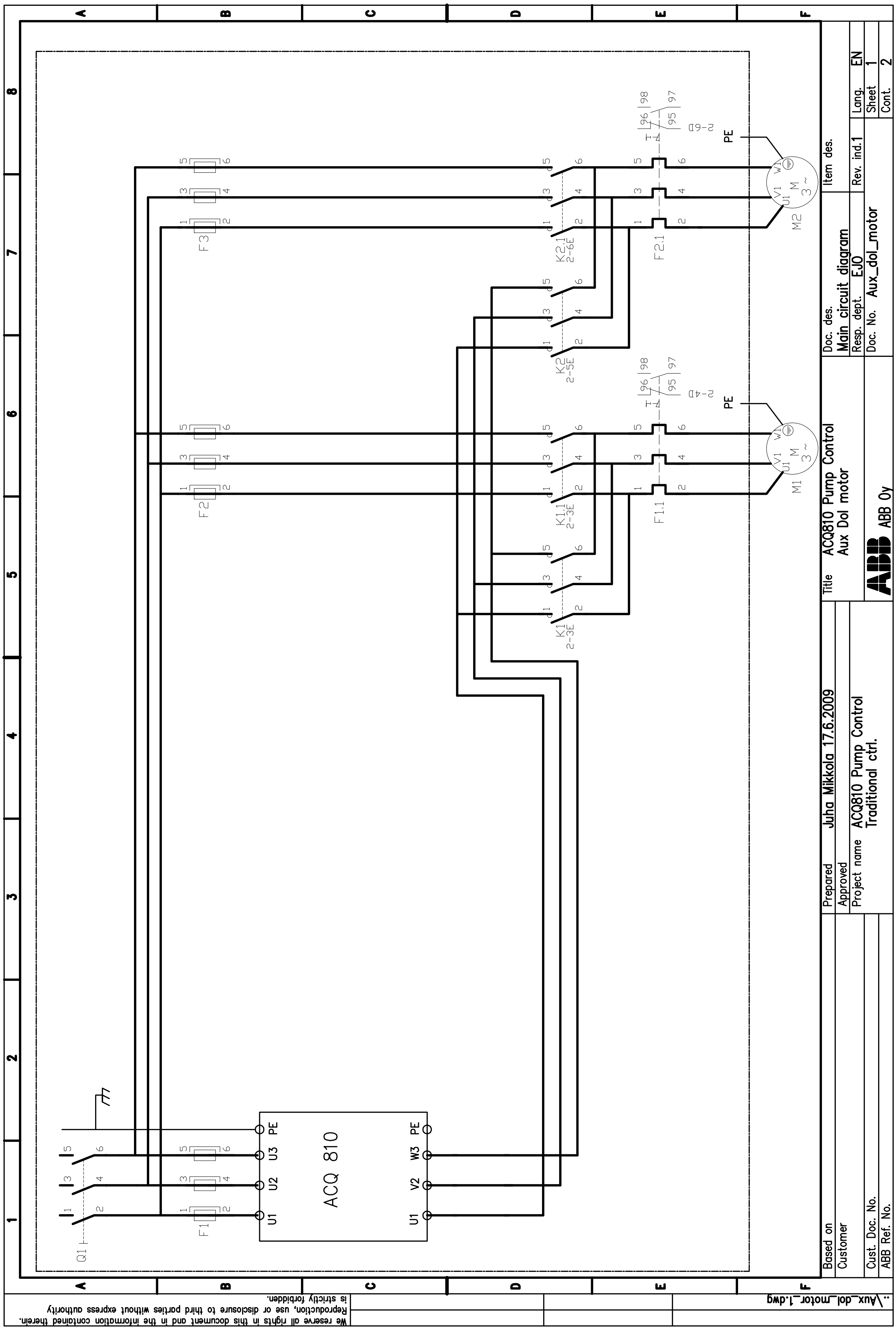

CONTROL VOLTAGE
ACQ 810
CONTROL CARD (JCON)
RO1
RO2
24VDC
INTERLOCK M2
DI2
INTERLOCK M3
DI4
START CTRL
DI1
AI2
PI PRESSURE FEEDBACK
0–10 bar / 4–20 mA
OUTLET
S1
S2
S3
K2
K3
K2.3
K3.3
F2.1
F3.1
M2/DOL
M3/DOL
S1,S2
A = AUTO
H = HAND
1 = START
Based on
Customer
Cust. Doc. No.
ABB Ref. No.
Prepared
Juha Mikkola 17.6.2009
Approved
Project name
ACQ 810 Pump Control
Traditional ctrl.
Title
ACQ810 Pump Control
Aux Dol motor
ABB Oy
Doc. des.
Circuit diagram
Resp. dept.
EJO
Doc. No.
Aux_dol_motor
Item des.
Rev. ind.1
Lang.
EN
Sheet
2
Cont.
–
We reserve all rights in this document and in the information contained therein.
Reproduction, use or disclosure to third parties without express authority
is strictly forbidden.
:\Aux_dol_motor_2.dwg

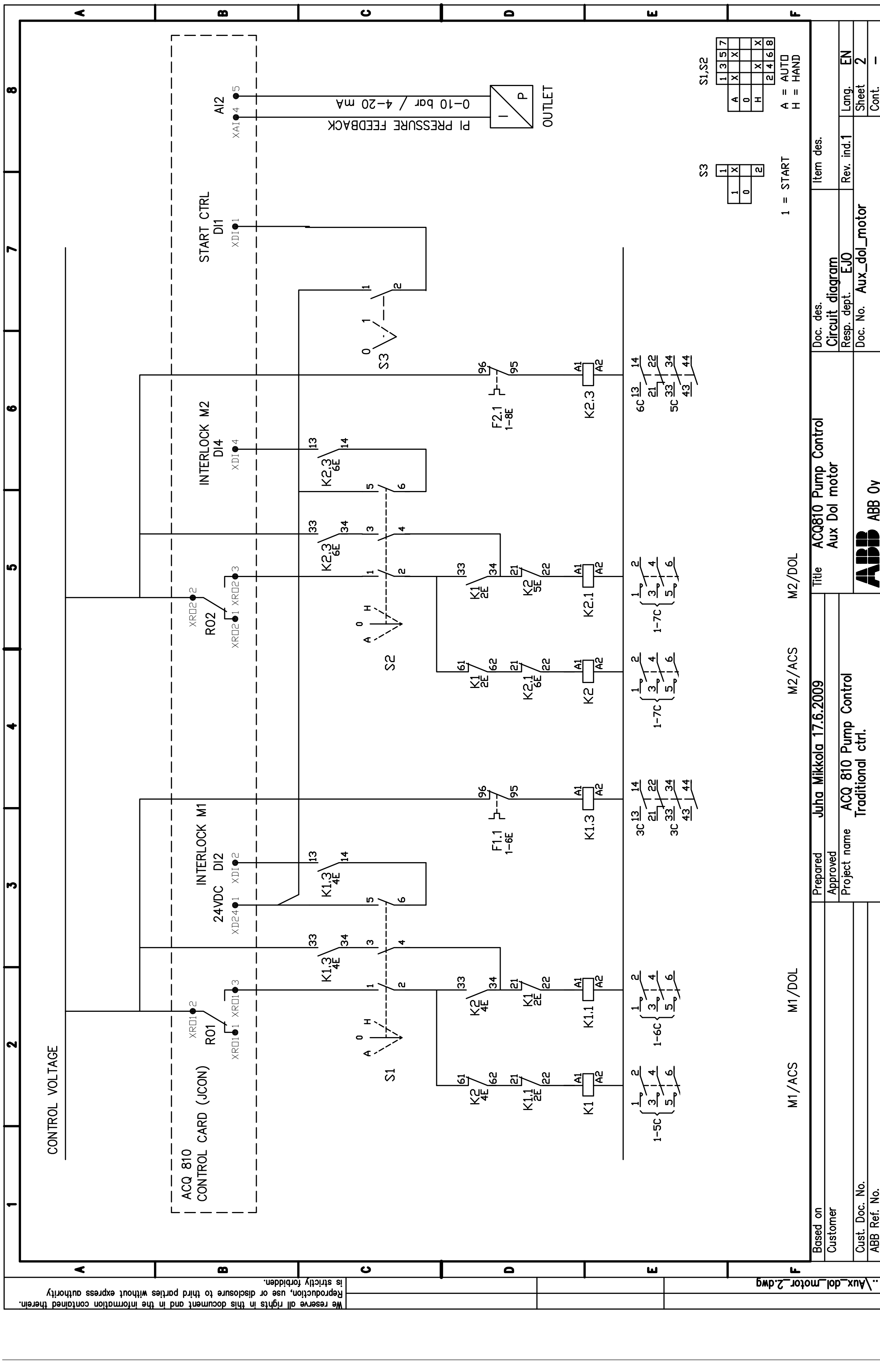
## Pump alternation using contactors

In this example, two pumps both have a contactor configuration that enables them to be connected either to the drive output or the supply. At any given time, one pump is connected to the drive, the other is connected to the supply.

Below is a listing of typical parameter values that would be used in this configuration.

| **Parameter** | | **Trad. pump control macro default** |
|---|---|---|
| **No.** | **Name** | |
| *14.42* | *RO1 src* | *Trad pump1* |
| *14.45* | *RO2 src* | *Trad pump2* |
| *16.20* | *Macro selected* | *Trad ctrl* |
| *26.02* | *Const speed sel1* | C.FALSE |
| *75.01* | *Operation mode* | *Trad ctrl* |
| *75.02* | *Nbr of pumps* | 2 |
| *75.25* | *Drive start dly* | 1 s |
| *78.01* | *Autochg style* | *All stop* |
| *78.02* | *Autochg trad* | *All* |
| *78.03* | *Interlock mode* | *On* |
| *78.06* | *Interlock pump 1* | *DI2* |
| *78.07* | *Interlock pump 2* | *DI4* |
| *78.07* | *Interlock pump 3* | *Not used* |


ACQ 810
Q1
F1
F2
F3
K1
K1.1
K2
K2.1
F1.1
F2.1
M1
M2
PE
Title ACQ810 Pump Control
Aux Dol motor
ABB Oy
Doc. des.
Main circuit diagram
Resp. dept. EJO
Doc. No. Aux_dol_motor
Item des.
Rev. ind.1
Lang. EN
Sheet 1
Cont. 2
Prepared Juha Mikkola 17.6.2009
Approved
Project name ACQ810 Pump Control
Traditional ctrl.
Based on
Customer
Cust. Doc. No.
ABB Ref. No.
We reserve all rights in this document and in the information contained therein. Reproduction, use or disclosure to third parties without express authority is strictly forbidden.
..\Aux_dol_motor_1.dwg


ACQ 810
CONTROL CARD (JCON)
CONTROL VOLTAGE
INTERLOCK M1
INTERLOCK M2
START CTRL
PI PRESSURE FEEDBACK
0-10 bar / 4-20 mA
OUTLET
1 = START
A = AUTO
H = HAND
M1/ACS
M1/DOL
M2/ACS
M2/DOL
ACQ810 Pump Control
Aux Dol motor
ACQ 810 Pump Control
Traditional ctrl.
Juha Mikkola 17.6.2009
Circuit diagram
Aux_dol_motor
ABB Oy
We reserve all rights in this document and in the information contained therein. Reproduction, use or disclosure to third parties without express authority is strictly forbidden.

# Level control macro

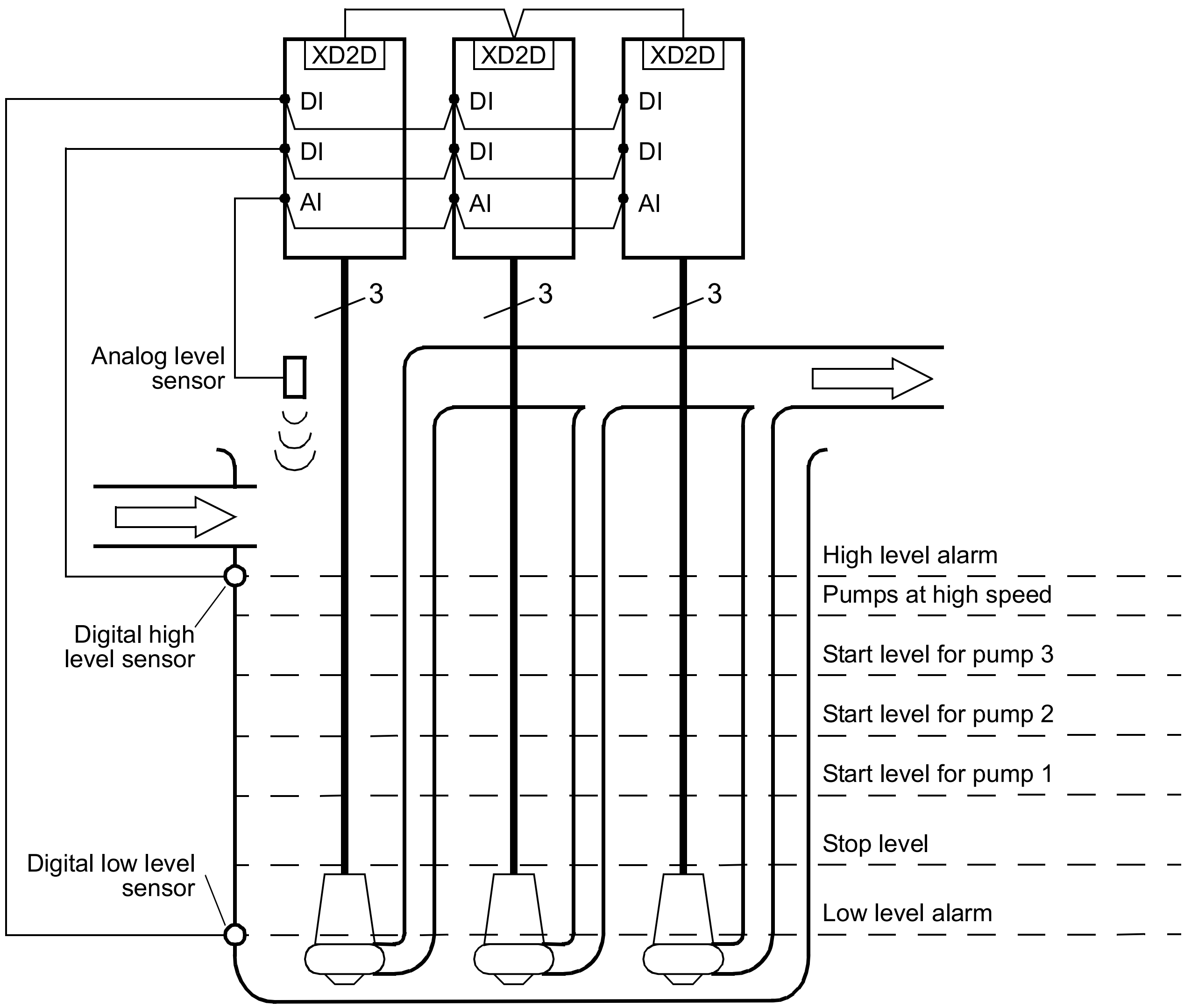
## Description and typical application

The level control macro is designed for controlling a station of 1 to 8 pumps that is used for either emptying or filling a container.

The level control functionality is activated by setting parameter *79.01 Level mode* to *Emptying* or *Filling* and selecting external control location EXT2. The start levels for the pumps (as well as the alarm levels) are set by parameters in group *79 Level control*.

At any given time, one of the drives acts as master. The master status can be rotated between the drives using the Autochange function, or one drive can be fixed to master status. The start/stop level settings of the master are the ones in effect.

The following drawing represents a station with three submersible pumps in emptying mode. Each pump has a pre-defined start level, and more pumps are started as the level in the container rises. The level sensor is connected to an analog input that is selected as the process actual value in parameter group *28 Procact sel*.



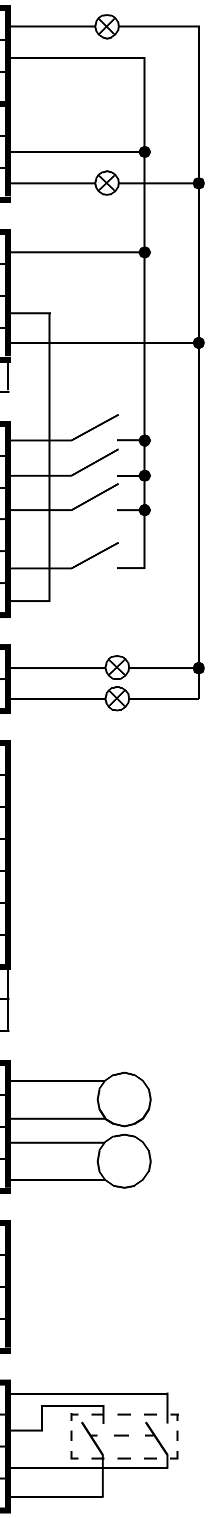
## Default settings

Below is a listing of default parameter values that differ from those listed in chapter *Additional parameter data* (page *287*).

| Parameter | | Level control macro default | |
|---|---|---|---|
| **No.** | **Name** | **Single pump** | **Multiple pumps** |
| *16.16* | *Menu set active* | *Level short* | *M lvl short* |
| *16.20* | *Macro selected* | *Level ctrl* | *Multi level* |
| *76.01* | *Enable MF comm* | *No* | *Yes* |
| *77.01* | *Sleep mode sel* | *Not used* | *Not used* |
| *79.01* | *Level mode* | *Emptying* | *Emptying* |

## Default control connections

| | | |
|---|---|---|
| | | **XPOW** |
| External power input<br>24 V DC, 1.6 A | +24VI | 1 |
| | GND | 2 |
| | | **XRO1, XRO2** |
| Relay output RO1 [Ready]<br>250 V AC / 30 V DC<br>2 A | NO | 1 |
| | COM | 2 |
| | NC | 3 |
| Relay output RO2 [Fault(-1)]<br>250 V AC / 30 V DC<br>2 A | NO | 4 |
| | COM | 5 |
| | NC | 6 |
| | | **XD24** |
| +24 V DC | +24VD | 1 |
| Digital input ground | DIGND | 2 |
| +24 V DC | +24VD | 3 |
| Digital input/output ground | DIOGND | 4 |
| Ground selection jumper | | |
| | | **XDI** |
| Digital input DI1 [Stop/Start] | DI1 | 1 |
| Digital input DI2 [Constant speed 1] | DI2 | 2 |
| Digital input DI3 [Reset] | DI3 | 3 |
| Digital input DI4 | DI4 | 4 |
| Digital input DI5 [EXT1/EXT2 selection] | DI5 | 5 |
| Start interlock (0 = Stop) | DIIL | A |
| | | **XDIO** |
| Digital input/output DIO1 [Output: Ready] | DIO1 | 1 |
| Digital input/output DIO2 [Output: Running] | DIO2 | 2 |
| | | **XAI** |
| Reference voltage (+) | +VREF | 1 |
| Reference voltage (–) | -VREF | 2 |
| Ground | AGND | 3 |
| Analog input AI1 (Current or voltage, selectable by jumper AI1)<br>[Current] [Speed reference 1] | AI1+ | 4 |
| | AI1- | 5 |
| Analog input AI2 (Current or voltage, selectable by jumper AI2)<br>[Current] [Process actual value 1*] | AI2+ | 6 |
| | AI2- | 7 |
| AI1 current/voltage selection jumper | | AI1 |
| AI2 current/voltage selection jumper | | AI2 |
| | | **XAO** |
| Analog output AO1 [Current] | AO1+ | 1 |
| | AO1- | 2 |
| Analog output AO2 [Speed rpm] | AO2+ | 3 |
| | AO2- | 4 |
| | | **XD2D** |
| Drive-to-drive link termination jumper | | T |
| Drive-to-drive link. | B | 1 |
| | A | 2 |
| | BGND | 3 |
| | | **XSTO** |
| Safe torque off. Both circuits must be closed for the drive to start. | OUT1 | 1 |
| | OUT2 | 2 |
| | IN1 | 3 |
| | IN2 | 4 |
| Control panel connection | | |
| Memory unit connection | | |

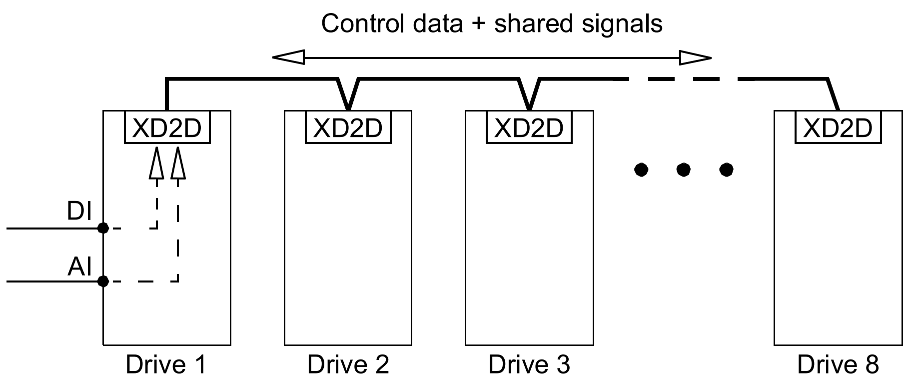


# Multipump control macro

## Description and typical application

This macro is suitable for pump stations that consist of multiple pumps, each controlled by a separate drive.

The configuration supports redundancy so that in case of a pump failure or maintenance action on one drive, the remaining drives continue operation. The drives communicate with each other through the drive-to-drive (D2D) link. It is possible to distribute two analog and five digital signals from a specific drive to the other drives via the drive-to-drive link (see parameters *76.11*…*76.16*).




The multipump macro has three modes selectable by parameter.

- In master-regulated operation, when the load increases, the master's speed increases. After the master has reached full speed, other drives are started one by one. Depending on a parameter setting, the master status is retained by the first drive, or passed on to the drive that was started last.
- Follower drives are run either at a pre-set speed (i.e. at the optimal operating point of the pump) or at the same speed as the master. In both these modes, drives can be prioritized so that the one with the highest priority is the first to be started.
- In direct follower operation, all drives run in synchronization with the master. This mode can be used in time-critical applications or for testing of the pump installation.

## Default settings

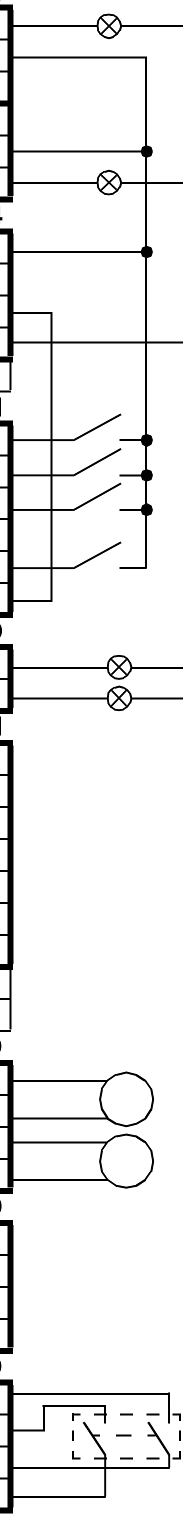
Below is a listing of default parameter values that differ from those listed in chapter *Additional parameter data* (page *287*).

| **Parameter** | | **Multipump control macro default** |
|---|---|---|
| **No.** | **Name** | |
| *16.16* | *Menu set active* | *M pump short* |
| *16.20* | *Macro selected* | *Multi pump* |

| | | |
|---|---|---|
| *75.01* | *Operation mode* | *Multipump* |
| *76.01* | *Enable MF comm* | *Yes* |

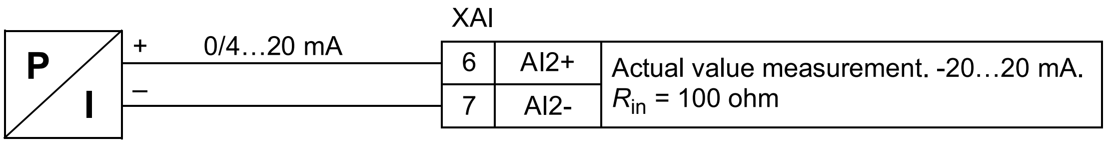
## Default control connections

| | | |
|---|---|---|
| | | **XPOW** |
| External power input<br>24 V DC, 1.6 A | +24VI | 1 |
| | GND | 2 |
| | | **XRO1, XRO2** |
| Relay output RO1 [Ready]<br>250 V AC / 30 V DC<br>2 A | NO | 1 |
| | COM | 2 |
| | NC | 3 |
| Relay output RO2 [Fault(-1)]<br>250 V AC / 30 V DC<br>2 A | NO | 4 |
| | COM | 5 |
| | NC | 6 |
| | | **XD24** |
| +24 V DC | +24VD | 1 |
| Digital input ground | DIGND | 2 |
| +24 V DC | +24VD | 3 |
| Digital input/output ground | DIOGND | 4 |
| Ground selection jumper | | |
| | | **XDI** |
| Digital input DI1 [Stop/Start] | DI1 | 1 |
| Digital input DI2 [Constant speed 1] | DI2 | 2 |
| Digital input DI3 [Reset] | DI3 | 3 |
| Digital input DI4 | DI4 | 4 |
| Digital input DI5 [EXT1/EXT2 selection] | DI5 | 5 |
| Start interlock (0 = Stop) | DIIL | A |
| | | **XDIO** |
| Digital input/output DIO1 [Output: Ready] | DIO1 | 1 |
| Digital input/output DIO2 [Output: Running] | DIO2 | 2 |
| | | **XAI** |
| Reference voltage (+) | +VREF | 1 |
| Reference voltage (–) | -VREF | 2 |
| Ground | AGND | 3 |
| Analog input AI1 (Current or voltage, selectable by jumper AI1) [Current] [Speed reference 1] | AI1+ | 4 |
| | AI1- | 5 |
| Analog input AI2 (Current or voltage, selectable by jumper AI2) [Current] [Process actual value 1*] | AI2+ | 6 |
| | AI2- | 7 |
| AI1 current/voltage selection jumper | | AI1 |
| AI2 current/voltage selection jumper | | AI2 |
| | | **XAO** |
| Analog output AO1 [Current] | AO1+ | 1 |
| | AO1- | 2 |
| Analog output AO2 [Speed rpm] | AO2+ | 3 |
| | AO2- | 4 |
| | | **XD2D** |
| Drive-to-drive link termination jumper | | T |
| Drive-to-drive link. | B | 1 |
| | A | 2 |
| | BGND | 3 |
| | | **XSTO** |
| Safe torque off. Both circuits must be closed for the drive to start. | OUT1 | 1 |
| | OUT2 | 2 |
| | IN1 | 3 |
| | IN2 | 4 |
| Control panel connection | | |
| Memory unit connection | | |

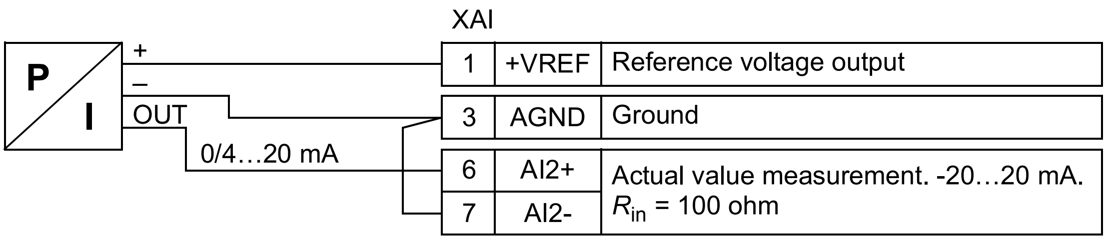


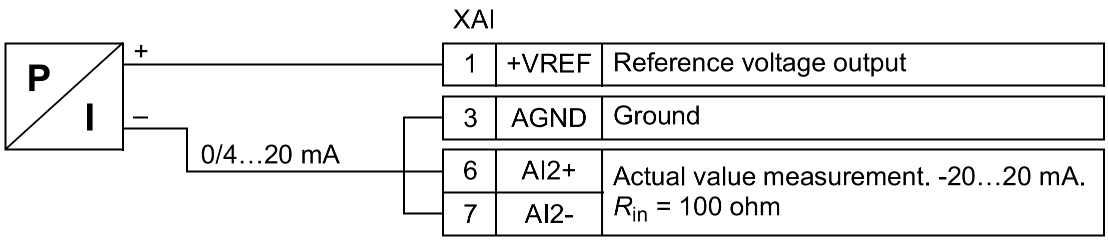
*See *Pressure sensor connection examples* on page *108*.

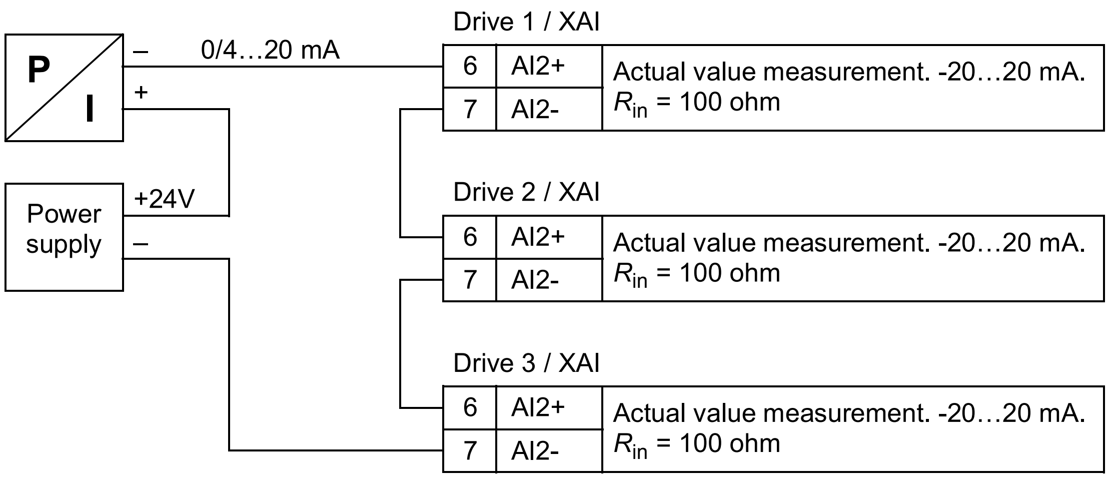
# Pressure sensor connection examples




**Note:** The sensor must be powered externally.



6

# Parameters

## 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 *16.21 Menu selection* to *Full*.

## 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 | 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).<br>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.<br>A pointer value is given in the format **P.xx.yy.zz**, where **xx** = parameter group, **yy** = parameter index, **zz** = bit number.<br>Pointing to a nonexisting bit will be interpreted as 0 (FALSE).<br>In addition to the “Const” and “Pointer” selections, bit pointer settings may also have other pre-selected settings. |
| 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 | A parameter that points to the value of another actual signal or parameter.<br>A pointer value is given in the format **P.xx.yy**, where **xx** = parameter group, **yy** = parameter index. |

## Summary of parameter groups

# Parameter listing

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| **01 Actual values** | | Basic signals for monitoring of the drive. | |
| 01.01 | Motor speed rpm | Filtered, estimated motor speed in rpm. 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 |
| 01.14 | 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% |
| 01.17 | Motor temp1 | Measured temperature of motor 1 in degrees Celsius when a KTY 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 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 SupplyVoltAutoId*, the automatically determined supply voltage. | 10 = 1 V |
| 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. Can be reset by entering a 0 using the DriveStudio PC tool. | 1 = 1 kWh |
| 01.25 | kWh supply | Amount of energy that the drive has taken from the AC supply in kilowatt-hours. Can be reset by entering a 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 a 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 a 0 using the DriveStudio PC tool.<br>**Note:** The drive logic uses this value for equalization of pump running duties. See section *Autochange* (page *61*). | 1 = 1 h |
| 01.28 | Fan on-time | Running time of the drive cooling fan. Can be reset by entering a 0 using the DriveStudio PC tool. | 1 = 1 h |
| 01.29 | Torq nom scale | Nominal torque which corresponds to 100%.<br>**Note:** This value is copied from parameter *99.12 Mot nom torque* if entered. Otherwise the value is calculated. | 1000 = 1 N•m |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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.<br>See parameter group *23 Speed ctrl* on page *176*. | 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 of fault limit. | 10 = 1% |
| 01.35 | Saved energy | Energy saved in kWh compared to direct-on-line motor connection.<br>**Note:** This value is derived from subtracting the drive's energy consumed from the direct-on-line consumption calculated on the basis of parameter *45.08 Pump ref power*. As such, the accuracy of this signal is dependent on the accuracy of the direct-on-line power estimate entered in that parameter.<br>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*.<br>See parameter group *45 Energy optimising* on page *227*. | 100 = 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).<br>See parameter group *45 Energy optimising* on page *227*. | 10 = 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 states and values; control and status words. | |
| 02.01 | DI status | Status of digital inputs DI6…DI1. **Example:** 000001 = DI1 is on, DI2…DI6 are off. | - |
| 02.02 | RO status | Status of relay outputs RO5…RO1. **Example:** 00001 = RO1 is energized, RO2…RO5 are de-energized. | - |
| 02.03 | DIO status | Status of digital input/outputs DIO4…DIO1. **Example:** 0000001001 = DIO1 and DIO4 are on, remainder are off.<br>DIO3…DIO4 are available only with an FIO I/O extension module. | - |
| 02.04 | AI1 | Value of analog input AI1 in V or mA. Input type is selected with a jumper on the JCU Control Unit. | 1000 = 1 unit |
| 02.05 | AI1 scaled | Scaled value of analog input AI1. See parameters *13.04 AI1 max scale* and *13.05 AI1 min scale*. | 1000 = 1 unit |
| 02.06 | AI2 | Value of analog input AI2 in V or mA. Input type is selected with a jumper on the JCU Control Unit. | 1000 = 1 unit |
| 02.07 | AI2 scaled | Scaled value of analog input AI2. See parameters *13.09 AI2 max scale* and *13.10 AI2 min scale*. | 1000 = 1 unit |
| 02.08 | AI3 | Value of analog 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 analog input AI3. See parameters *13.14 AI3 max scale* and *13.15 AI3 min scale*. | 1000 = 1 unit |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 02.10 | AI4 | Value of analog 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 analog input AI4. See parameters *13.19 AI4 max scale* and *13.20 AI4 min scale*. | 1000 = 1 unit |
| 02.12 | AI5 | Value of analog 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 analog input AI5. See parameters *13.24 AI5 max scale* and *13.25 AI5 min scale*. | 1000 = 1 unit |
| 02.16 | AO1 | Value of analog output AO1 in mA. | 1000 = 1 mA |
| 02.17 | AO2 | Value of analog output AO2 in mA. | 1000 = 1 mA |
| 02.18 | AO3 | Value of analog output AO3 in mA. | 1000 = 1 mA |
| 02.19 | AO4 | Value of analog 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 *Freq output*). | 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 *357*.<br>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…10 | Reserved | | | | |
| 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.<br>• Local fieldbus: Transfer to fieldbus local control (control through Control Word or reference). Fieldbus steals the control.<br>• 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 *357*. | - |

| 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 | Relay 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 *313* |
| | | 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 *313*. |
| | | 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* and the user manual of the fieldbus adapter. | |
| 29 | SW B29 | | |
| 30 | SW B30 | | |
| 31 | SW B31 | | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 02.26 | FBA main ref1 | Internal and scaled reference 1 of the drive received through the fieldbus adapter interface. See parameter *50.04 FBA ref1 modesel* and chapter *Control through a fieldbus adapter* on page *357*. | 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 FBA ref2 modesel* and chapter *Control through a fieldbus adapter* on page *357*. | 1 = 1 |
| 02.34 | Panel ref | Reference given from the control panel. See also parameter *56.07 Local ref unit*. | 100 = 1 rpm<br>10 = 1% |

| 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 *329*.<br>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…10 | Reserved | | | | |
| 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.<br>• Local fieldbus: Transfer to fieldbus local control (control through Control Word or reference). Fieldbus steals the control.<br>• 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 *329*. | - |

| 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 *313*. |
| | | 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 *313*. |
| | | 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 FBA ref1 modesel* and chapter *Control through the embedded fieldbus interface* on page *329*. | - |
| 02.39 | EFB main ref2 | Internal and scaled reference 2 of the drive received through the embedded fieldbus interface. See parameter *50.05 FBA ref2 modesel* and chapter *Control through the embedded fieldbus interface* on page *329*. | -- |
| 02.40 | FBA setpoint | Target parameter for writing the setpoint from the fieldbus. The unit and scaling are defined by parameters *28.06 Act unit sel* and *28.07 Act FBA scaling* respectively. | - |
| 02.41 | FBA act val | Target parameter for writing a feedback value from the fieldbus. The unit and scaling are defined by parameters *28.06 Act unit sel* and *28.07 Act FBA scaling* respectively. | - |
| 02.42 | Shared DI | Status of shared digital inputs received through the drive-to-drive link. **Example:** 00000001 = DI1 is on, DI2…DI6 are off. See parameters *76.11*…*76.16*. | - |
| 02.43 | Shared signal 1 | Shows the value of shared signal 1 as received through the drive-to-drive link. See parameters *76.11*…*76.16*. | - |
| 02.44 | Shared signal 2 | Shows the value of shared signal 2 as received through the drive-to-drive link. See parameters *76.11*…*76.16*. | - |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| **03 Control values** | | Speed and torque control values. | |
| 03.03 | SpeedRef unramp | Used speed reference ramp input 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% |
| 03.13 | 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% |
| 03.17 | Flux actual | Actual flux reference in percent. | 1 = 1% |
| 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 | Act val | Final actual value after selection (see parameter group *28 Procact sel*). The unit and scaling are defined by parameters *28.06 Act unit sel* and *28.07 Act FBA scaling* respectively.<br>See also parameters *04.20…04.22*. | - |
| 04.02 | Setpoint | Final setpoint (reference) value after selection (see parameter group *29 Setpoint sel*). The unit and scaling are defined by parameters *28.06 Act unit sel* and *28.07 Act FBA scaling* respectively.<br>See also parameters *04.23…04.25*. | - |
| 04.04 | Process PID err | Process PID error, i.e. difference between PID setpoint and actual value. | 10 = 1% |
| 04.05 | Process PID out | Output of the process PID controller. | 10 = 1% |
| 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 |
| 04.20 | Act val 1 out | Actual value 1 (selected by parameter *28.02 Act val 1 src*). | 100 = 1 unit |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 04.21 | Act val 2 out | Actual value 2 (selected by parameter *28.03 Act val 2 src*). | 100 = 1 unit |
| 04.22 | Act val % | Final actual value in %. | 100 = 1% |
| 04.23 | Setpoint val 1 | Setpoint 1 (selected by parameter *29.02 Setpoint 1 src*). | 100 = 1 unit |
| 04.24 | Setpoint val 2 | Setpoint 2 (selected by parameter *29.03 Setpoint 2 src*). | 100 = 1 unit |
| 04.25 | Setpoint val % | Final setpoint in %. | 100 = 1% |
| 04.26 | Wake up level | Final calculated wake-up level. See the selections of parameter *77.08 Wake up mode sel*. | 100 = 1 |
| 04.27 | Shared source | Node number of the drive that is currently the source of shared signals. See parameters *76.11*…*76.16*. | TBA |
| 04.28 | Pump runtime | Pump run-time counter. The counter runs when the drive is running (started). Can be reset using parameter *78.14 Runtime change*. | 1 = 1 h |
| 04.29 | Trad 1 runtime | Pump 1 run-time counter (for traditional control – see page *94*). Can be reset using parameter *78.14 Runtime change*. | 1 = 1 h |
| 04.30 | Trad 2 runtime | Pump 2 run-time counter (for traditional control – see page *94*). Can be reset using parameter *78.14 Runtime change*. | 1 = 1 h |
| 04.31 | Trad 3 runtime | Pump 3 run-time counter (for traditional control – see page *94*). Can be reset using parameter *78.14 Runtime change*. | 1 = 1 h |
| 04.32 | Trad 4 runtime | Pump 4 run-time counter (for traditional control – see page *94*). Can be reset using parameter *78.14 Runtime change*. | 1 = 1 h |
| 04.33 | Trad 5 runtime | Pump 5 run-time counter (for traditional control – see page *94*). Can be reset using parameter *78.14 Runtime change*. | 1 = 1 h |
| 04.34 | Trad 6 runtime | Pump 6 run-time counter (for traditional control – see page *94*). Can be reset using parameter *78.14 Runtime change*. | 1 = 1 h |
| 04.35 | Trad 7 runtime | Pump 7 run-time counter (for traditional control – see page *94*). Can be reset using parameter *78.14 Runtime change*. | 1 = 1 h |
| 04.36 | Trad 8 runtime | Pump 8 run-time counter (for traditional control – see page *94*). Can be reset using parameter *78.14 Runtime change*. | 1 = 1 h |

| | | | |
|---|---|---|---|
| **05 Pump values** | | Pump station actual values. | |
| 05.01 | MF status | State of drive in a multipump configuration (several drives connected by the drive-to-drive link). | |
| | No | Drive-to-drive communication is not active. | 0 |
| | Standby | The drive is ready to start and waiting for a start command from the master. | 1 |
| | Master | The drive is running and currently the master. | 2 |
| | Follower | The drive is running and currently a follower. | 3 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 05.02 | Trad pump cmd | Pump control word. The bits of this parameter can be used to control the relay outputs that switch pumps on and off. | - |

| Bit | Name |
|---|---|
| 0 | Pump 1. **Note:** The setting of parameter *78.02 Autochg trad* determines whether “Pump 1” refers to the first pump of the station, or the first auxiliary pump of the station. |
| 1 | Pump 2 |
| 2 | Pump 3 |
| 3 | Pump 4 |
| 4 | Pump 5 |
| 5 | Pump 6 |
| 6 | Pump 7 |
| 7 | Pump 8 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 05.03 | Trad master | In traditional pump control, the number of the pump that is directly controlled by the drive. | 1 = 1 |
| 05.04 | Nbr aux pumps on | Number of auxiliary pumps running. | 1 = 1 |
| 05.05 | Flow act | Actual flow as calculated by the drive. See parameter group *80 Flow calculation* (page *267*). | 100 = 1 $m^3/h$ |
| 05.06 | Flow by head | Flow calculated on the basis of the HQ performance curve. See parameter group *80 Flow calculation* (page *267*). | 100 = 1 $m^3/h$ |
| 05.07 | Flow by power | Flow calculated on the basis of the PQ performance curve. See parameter group *80 Flow calculation* (page *267*). | 100 = 1 $m^3/h$ |
| 05.08 | Total flow | Total calculated flow. Stored when the drive is not powered. Can be reset using parameter *80.33 Sum flow reset*. | 1 = 1 $m^3$ |
| 05.09 | Bypass ref | Reference used when parameter *75.01 Operation mode* is set to *Reg bypass*. | 10 = 1 rpm |
| 05.10 | Speed ref | Final speed reference from the pump control logic. | 10 = 1 rpm |
| 05.20 | kWh current read | Energy consumed during the current period.<br>The length of the period is set by parameter *83.02 Mon period*. | 1 = 1 kWh |
| 05.21 | kWh prev read | Energy consumed during the last completed period. The length of the period is set by parameter *83.02 Mon period*. | 1 = 1 kWh |
| 05.22 | kWh posprev read | Energy consumed during the period before the last completed period. The length of the period is set by parameter *83.02 Mon period*. | 1 = 1 kWh |
| 05.23 | kWh cur mon read | Energy consumed during the current month. | 1 = 1 kWh |
| 05.24 | kWh January | Energy consumed during last January. | 1 = 1 kWh |
| 05.25 | kWh February | Energy consumed during last February. | 1 = 1 kWh |
| 05.26 | kWh March | Energy consumed during last March. | 1 = 1 kWh |
| 05.27 | kWh April | Energy consumed during last April. | 1 = 1 kWh |
| 05.28 | kWh May | Energy consumed during last May. | 1 = 1 kWh |
| 05.29 | kWh June | Energy consumed during last June. | 1 = 1 kWh |
| 05.30 | kWh July | Energy consumed during last July. | 1 = 1 kWh |
| 05.31 | kWh August | Energy consumed during last August. | 1 = 1 kWh |
| 05.32 | kWh September | Energy consumed during last September. | 1 = 1 kWh |
| 05.33 | kWh October | Energy consumed during last October. | 1 = 1 kWh |
| 05.34 | kWh November | Energy consumed during last November. | 1 = 1 kWh |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 05.35 | kWh December | Energy consumed during last December. | 1 = 1 kWh |
| 05.36 | First in order | The first pump in the current Autochange sequence. | 1 = 1 |
| 05.37 | Time autochg | Time elapsed since last Autochange. | 1 = 1 ms |
| 05.39 | Next start node | (Only valid when the drive is master.) Node number of the next drive to be started. | 1 = 1 |
| **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*. |
| | | 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*. |
| | | 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*. |
| 13…15 | 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 | Reserved | |
| 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 | Sleep active | 1 = Sleep mode active. |
| | | 0 = Sleep mode inactive. |
| 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 | 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 | Bal active | 1 = Speed controller output is being forced to value of parameter *27.35 PID bal ref*. |
| 5 | PI tune active | 1 = Speed controller autotuning procedure is active. |
| 6 | PI tune request | 1 = Speed controller autotuning has been requested by parameter *23.20 PI tune mode*. |
| 7 | PI tune done | 1 = Speed controller autotuning procedure has been completed successfully. |
| 8…15 | Reserved | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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, 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…4 | Reserved | |
| 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 maximum speed limit *20.02 Minimum speed*. |
| 7…15 | Reserved | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 06.07 | Torq lim status | Torque controller limitation status word. | - |

| Bit | Name | Information |
|---|---|---|
| 0 | Undervolt-age | 1 = Intermediate circuit DC undervoltage. * |
| 1 | Overvoltage | 1 = Intermediate circuit DC overvoltage. * |
| 2…3 | Reserved | |
| 4 | Internal cur-rent | 1 = An inverter current limit is active. The limit is identified by bits 8…11. |
| 5 | Reserved | |
| 6 | Motor pull-out | 1 = 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 maxi-mum | 1 = Inverter maximum output current limit is active (limits the drive output current $I_{MAX}$). ** |
| 10 | User cur-rent | 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. ** |
| 12…15 | Reserved | |
| * One of bits 0…3 can be on simultaneously. The bit typically indicates the limit that is exceeded first.<br>** Only one of bits 9…11 can be on simultaneously. The bit typically indicates the limit that is exceeded first. | | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 06.12 | Op mode ack | Operation mode acknowledge: 0 = Stopped, 1 = Speed, 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 *202*). | - |
| 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 *214*). 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. |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 06.20 | Pump status word | Pump status word. | - |

| Bit | Name | Value | Information |
|---|---|---|---|
| 0 | Trad | 1 | Traditional pump control mode active. |
| 1 | Reg bypass | 1 | PID controller bypass mode active. |
| 2 | Multi pump | 1 | Multipump functionality (via drive-to-drive link) active. |
| 3 | Level control | 1 | Level control active. |
| 4 | Sleep | 1 | Sleep mode active. |
| 5 | Boosting | 1 | Sleep boost active. |
| 6 | Pipe filling | 1 | Soft pipefill function active. |
| 7 | Bypass | 1 | TBA |
| 8 | Cleaning | 1 | Cleaning sequence active. |
| 9 | Analyzer ID run | 1 | Reserved |
| 10 | PID ref freeze | 1 | PID controller input frozen. |
| 11 | PID out freeze | 1 | PID controller output frozen. |
| 12 | Balancing | 1 | PID balancing reference in force. |
| 13 | No aux pumps | 1 | No auxiliary pumps available to be started. |
| 14 | Autochange | 1 | Autochange function active. |
| 15 | High prot spd | 1 | Outlet pressure monitoring: forced reference active. |
| 16 | Low prot spd | 1 | Inlet pressure monitoring: forced reference active. |
| 17 | Speed ref 2 act | 1 | Speed reference 2 active. |
| 18 | Ext2 Speed mode | 1 | Speed control selected for external control location EXT2 by parameter *12.05 Ext2 ctrl mode*. |
| 19…31 | Reserved | | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 06.21 | Level status | Level control status word. | - |

| Bit | Name | Value | Information |
|---|---|---|---|
| 0 | Low level 1 | 1 | Low level 1 reached. |
| 1 | Low level 2 | 1 | Low level 2 reached. |
| 2 | Stop level | 1 | Stop level reached. |
| 3 | Start level 1 | 1 | Start level 1 reached. |
| 4 | Start level 2 | 1 | Start level 2 reached. |
| 5 | Start level 3 | 1 | Start level 3 reached. |
| 6 | Start level 4 | 1 | Start level 4 reached. |
| 7 | Start level 5 | 1 | Start level 5 reached. |
| 8 | Start level 6 | 1 | Start level 6 reached. |
| 9 | Start level 7 | 1 | Start level 7 reached. |
| 10 | Start level 8 | 1 | Start level 8 reached. |
| 11 | High level 1 | 1 | High level 1 reached. |
| 12 | High level 2 | 1 | High level 2 reached. |
| 13 | High speed | 1 | High speed reached. |
| 14…31 | Reserved | | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 06.22 | MF status word | Multipump communication status word. | - |

| Bit | Name | Value | Information |
|---|---|---|---|
| 0 | Master | 1 | Drive is master. |
| 1 | Follower | 1 | Drive is follower. |
| 2 | Master running | 1 | Master drive is running. |
| 3 | Copy of mstr | 1 | Sync mode active (par. *75.03* is set to *Copy of mstr*). |
| 4 | Node 1 | 1 | (Only valid if the drive is master.) Drive with node number 1 is present on the drive-to-drive link. |
| 5 | Node 2 | 1 | (Only valid if the drive is master.) Drive with node number 2 is present on the drive-to-drive link. |
| 6 | Node 3 | 1 | (Only valid if the drive is master.) Drive with node number 3 is present on the drive-to-drive link. |
| 7 | Node 4 | 1 | (Only valid if the drive is master.) Drive with node number 4 is present on the drive-to-drive link. |
| 8 | Node 5 | 1 | (Only valid if the drive is master.) Drive with node number 5 is present on the drive-to-drive link. |
| 9 | Node 6 | 1 | (Only valid if the drive is master.) Drive with node number 6 is present on the drive-to-drive link. |
| 10 | Node 7 | 1 | (Only valid if the drive is master.) Drive with node number 7 is present on the drive-to-drive link. |
| 11 | Node 8 | 1 | (Only valid if the drive is master.) Drive with node number 8 is present on the drive-to-drive link. |
| 12…31 | Reserved | | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| **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 |
| 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. Can be reset by entering a 0. | - |

| Bit | Name |
|---|---|
| 0…2 | Reserved |
| 3 | *SAFE TORQUE OFF* (page *314*) |
| 4 | *STO MODE CHANGE* (page *314*) |
| 5 | *MOTOR TEMPERATURE* (page *314*) |
| 6 | *EMERGENCY OFF* (page *314*) |
| 7 | *RUN ENABLE* (page *314*) |
| 8 | *ID-RUN* (page *315*) |
| 9 | *EMERGENCY STOP* (page *315*) |
| 10…12 | Reserved |
| 13 | *DEVICE OVERTEMP* (page *315*) |
| 14 | *INTBOARD OVERTEMP* (page *315*) |
| 15 | Reserved |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 08.06 | Alarm word2 | Alarm word 2. Can be reset by entering a 0. | - |

| Bit | Name |
|---|---|
| 0 | Reserved |
| 1 | *FIELDBUS COMM* (page *315*) |
| 2 | *LOCAL CTRL LOSS* (page *315*) |
| 3 | *AI SUPERVISION* (page *315*) |
| 4 | *FB PAR CONF* (page *315*) |
| 5 | *NO MOTOR DATA* (page *315*) |
| 6…15 | Reserved |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 08.07 | Alarm word3 | Alarm word 3. Can be reset by entering a 0. | - |

| Bit | Name |
|---|---|
| 0…2 | Reserved |
| 3 | *PS COMM* (page *316*) |
| 4 | *RESTORE* (page *316*) |
| 5 | *CUR MEAS CALIBRATION* (page *316*) |
| 6 | Reserved |
| 7 | *EARTH FAULT* (page *316*) |
| 8 | *AUTORESET* (page *316*) |
| 9 | *MOTOR NOM VALUE* (page *316*) |
| 10 | Reserved |
| 11 | *STALL* (page *316*) |
| 12 | *LCURVE* (page *316*) |
| 13 | *LCURVE PAR* (page *316*) |
| 14 | *FLUX REF PAR* (page *316*) |
| 15 | Reserved |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 08.08 | Alarm word4 | Alarm word 4. Can be reset by entering a 0. | - |

| Bit | Name |
|---|---|
| 0 | *OPTION COMM LOSS* (page *316*) |
| 1 | *SOLUTION ALARM* (page *320*) |
| 2 | *MOTTEMPAL2* (page *317*) |
| 3 | *IGBTOLALARM* (page *317*) |
| 4 | *IGBTTEMPALARM* (page *317*) |
| 5 | *COOLALARM* (page *317*) |
| 6 | *MENU CHANGED* (page *317*) |
| 7 | *TEMP MEAS FAILURE* (page *318*) |
| 8 | Maintenance counter alarms 2055…2071 (page *318*) |
| 9 | *DC NOT CHARGED* (page *318*) |
| 10 | *AUTOTUNE FAILED* (page *318*) |
| 11 | *START INTERLOCK* (page *318*) |
| 12 | *EFB COMM LOSS* (page *319*) |
| 13…15 | Reserved |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 08.09 | Alarm word5 | Alarm word 5. Can be reset by entering a 0. | - |

| Bit | Name |
|---|---|
| 0 | *PIPEFILL TIMEOUT* (page *319*) |
| 1 | *MIN FLOW* (page *319*) |
| 2 | *MAX FLOW* (page *319*) |
| 3 | *LOW PRESSURE* (page *319*) |
| 4 | *HIGH PRESSURE* (page *319*) |
| 5 | *VERY LOW PRESS* (page *319*) |
| 6 | *VERY HIGH PRESS* (page *319*) |
| 7 | *PROFILE HIGH* (page *319*) |
| 8 | *MAX CLEANINGS* (page *319*) |
| 9 | *ALL PUMPS INLOCKD* (page *319*) |
| 10 | *ENERGY LIMIT* (page *319*) |
| 11 | *DATE WRONG* (page *320*) |
| 12…13 | Reserved |
| 14 | *BOOSTING* (page *320*) |
| 15 | *PIPE FILLING* (page *320*) |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 08.10 | Alarm word6 | Alarm word 6. Can be reset by entering a 0. | - |

| Bit | Name |
|---|---|
| 0 | *NO MORE PUMPS* (page *320*) |
| 1 | *CLEANING* (page *320*) |
| 2 | *AUTOCHANGE* (page *320*) |
| 3 | *SLEEPING* (page *320*) |
| 4 | *START DELAY* (page *320*) |
| 5 | *LC TANK FULL* (page *320*) |
| 6 | *LC TANK EMPTY* (page *320*) |
| 7 | *MF MASTER LOST* (page *320*) |
| 8 | *MF NO SHARED DATA* (page *320*) |
| 9…15 | Reserved |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 08.20 | Pump fault word | Pump fault word. | - |

| Bit | Name |
|---|---|
| 0 | *MIN FLOW* (page *328*) |
| 1 | *MAX FLOW* (page *328*) |
| 2 | *LOW PRESSURE* (page *328*) |
| 3 | *HIGH PRESSURE* (page *328*) |
| 4 | *VERY LOW PRESS* (page *328*) |
| 5 | *VERY HIGH PRESS* (page *328*) |
| 6 | *MAX CLEANINGS* (page *328*) |
| 7 | *PIPEFILL TOUT* (page *328*) |
| 8 | *MF MASTER LOST* (page *328*) |
| 9 | *MF NO SHARED DATA* (page *328*) |
| 10…31 | Reserved |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 08.21 | Pump alarm word | Pump alarm word. | - |

| Bit | Name |
|---|---|
| 0 | *MIN FLOW* (page *319*) |
| 1 | *MAX FLOW* (page *319*) |
| 2 | *LOW PRESSURE* (page *319*) |
| 3 | *HIGH PRESSURE* (page *319*) |
| 4 | *VERY LOW PRESS* (page *319*) |
| 5 | *VERY HIGH PRESS* (page *319*) |
| 6 | *PROFILE HIGH* (page *319*) |
| 7 | *MAX CLEANINGS* (page *319*) |
| 8 | *CLEANING* (page *320*) |
| 9 | *PIPEFILL TIMEOUT* (page *319*) |
| 10 | *ALL PUMPS INLOCKD* (page *319*) |
| 11 | *ENERGY LIMIT* (page *319*) |
| 12 | *DATE WRONG* (page *320*) |
| 13…14 | Reserved |
| 15 | *BOOSTING* (page *320*) |
| 16 | *PIPE FILLING* (page *320*) |
| 17 | *NO MORE PUMPS* (page *320*) |
| 18 | *AUTOCHANGE* (page *320*) |
| 19 | *SLEEPING* (page *320*) |
| 20 | *START DELAY* (page *320*) |
| 21 | *MF MASTER LOST* (page *320*) |
| 22 | Reserved |
| 23 | *LC TANK FULL* (page *320*) |
| 24 | *LC TANK EMPTY* (page *320*) |
| 25 | *MF NO SHARED DATA* (page *320*) |
| 26…31 | Reserved |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| **09 System info** | | Drive type, program revision and option slot occupation information. | |
| 09.01 | Drive type | Displays the drive type (for example, ACQ810). | - |
| 09.02 | Drive rating ID | Displays the inverter type (ACQ810-…) of the drive.<br>0 = Unconfigured,<br>201 = 02A7-4, 202 = 03A0-4, 203 = 03A5-4, 204 = 04A9-4, 205 = 06A3-4, 206 = 08A3-4, 207 = 11A0-4, 208 = 14A4-4, 209 = 021A-4, 210 = 028A-4, 211 = 032A-4, 212 = 035A-4, 213 = 040A-4, 214 = 053A-4, 215 = 067A-4, 216 = 080A-4, 217 = 098A-4, 218 = 138A-4, 220 = 162A-4, 221 = 203A-4, 222 = 240A-4, 223 = 286A-4, 224 = 302A-4, 225 = 361A-4, 226 = 414A-4, 227 = 477A-4, 228 = 550A-4, 229 = 616A-4, 230 = 704A-4, 241 = 02A7-2, 242 = 03A0-2, 243 = 03A5-2, 244 = 04A9-2, 245 = 06A3-2, 246 = 08A3-2, 247 = 11A0-2, 248 = 14A4-2, 249 = 021A-2, 250 = 028A-2, 251 = 032A-2, 252 = 035A-2, 253 = 040A-2, 254 = 053A-2, 255 = 067A-2, 256 = 080A-2 | 1 = 1 |
| 09.03 | Firmware ID | Displays the firmware name. E.g. UIFQ. | - |
| 09.04 | Firmware ver | Displays the version of the firmware package in the drive, e.g. 2002 hex. | - |
| 09.05 | Firmware patch | Displays the version of the firmware patch in the drive. | 1 = 1 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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.<br>0 = No option, 1 = No comm, 2 = Unknown, 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-01, 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 *09.20 Option slot1*. | 1 = 1 |

| **10 Start/stop/dir** | | Start/stop/direction, run enable and emergency stop source selections; start inhibit and start interlock configuration. | |
|---|---|---|---|
| 10.01 | Ext1 start func | Selects the source of start and stop commands for external control location 1 (EXT1).<br>**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:<br><br>**State of source (via par *10.02*)** / **Command**:<br>0 -> 1: Start<br>1 -> 0: Stop | 1 |
| | 3-wire | 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:<br><br>**State of source 1 (via par. *10.02*)** / **State of source 2 (via par. *10.03*)** / **Command**:<br>0 -> 1 / 1 / Start<br>Any / 1 -> 0 / Stop<br>Any / 0 / Stop | 2 |
| | FBA | The start and stop commands are taken from the fieldbus Control Word selected by parameter *50.15 Fb cw used*. | 3 |
| | D2D | Reserved. | 4 |
| | In1F In2R | 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.<br><br>**State of source 1 (via par. *10.02*)** / **State of source 2 (via par. *10.03*)** / **Command**:<br>0 / 0 / Stop<br>1 / 0 / Start forward<br>0 / 1 / Start reverse<br>1 / 1 / Stop | 5 |

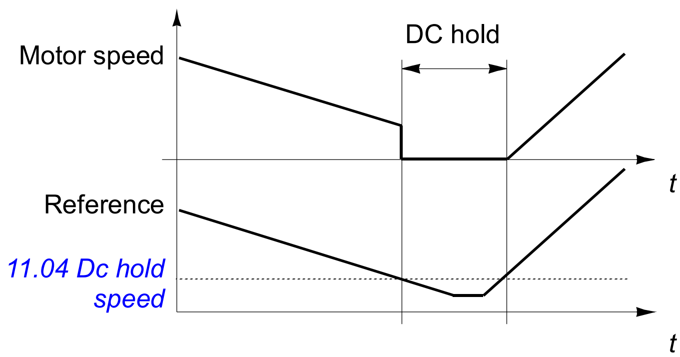
<table>
<tr><th>No.</th><th>Name/Value</th><th>Description</th><th>FbEq</th></tr>
<tr><td></td><td>In1St In2Dir</td><td>The source selected by 10.02 Ext1 start in1 is the start signal (0 = stop, 1 = start), the source selected by 10.03 Ext1 start in2 is the direction signal (0 = forward, 1 = reverse).</td><td>6</td></tr>
<tr><td>10.02</td><td>Ext1 start in1</td><td>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.<br>Note: This parameter cannot be changed while the drive is running.</td><td></td></tr>
<tr><td></td><td>DI1</td><td>Digital input DI1 (as indicated by 02.01 DI status, bit 0).</td><td>1073742337</td></tr>
<tr><td></td><td>DIO4</td><td>Digital input/output DIO4 (as indicated by 02.03 DIO status, bit 3).</td><td>1073938947</td></tr>
<tr><td></td><td>Timed func</td><td>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.</td><td>1074005518</td></tr>
<tr><td></td><td>Const</td><td rowspan="2">Constant and bit pointer settings (see Terms and abbreviations on page 109).</td><td rowspan="2">-</td></tr>
<tr><td></td><td>Pointer</td></tr>
<tr><td>10.03</td><td>Ext1 start in2</td><td>Selects source 2 of start and stop commands for external control location EXT1. See parameter 10.01 Ext1 start func, selection 3-wire.<br>Note: This parameter cannot be changed while the drive is running.</td><td></td></tr>
<tr><td></td><td>DI2</td><td>Digital input DI2 (as indicated by 02.01 DI status, bit 1).</td><td>1073807873</td></tr>
<tr><td></td><td>DI5</td><td>Digital input DI5 (as indicated by 02.01 DI status, bit 4).</td><td>1074004481</td></tr>
<tr><td></td><td>DIO5</td><td>Digital input/output DIO5 (as indicated by 02.03 DIO status, bit 4).</td><td>1074004483</td></tr>
<tr><td></td><td>Const</td><td rowspan="2">Bit pointer setting (see Terms and abbreviations on page 109).</td><td rowspan="2">-</td></tr>
<tr><td></td><td>Pointer</td></tr>
<tr><td>10.04</td><td>Ext2 start func</td><td>Selects the source of start and stop commands for external control location 2 (EXT2).<br>Note: This parameter cannot be changed while the drive is running.</td><td></td></tr>
<tr><td></td><td>Not sel</td><td>No start or stop command sources selected.</td><td>0</td></tr>
<tr><td></td><td>In1</td><td>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>
<tr><th>State of source (via par 10.05)</th><th>Command</th></tr>
<tr><td>0 -> 1</td><td>Start</td></tr>
<tr><td>1 -> 0</td><td>Stop</td></tr>
</table></td><td>1</td></tr>
<tr><td></td><td>3-wire</td><td>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>
<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>
<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>
</table></td><td>2</td></tr>
</table>

<table>
<tr><th>No.</th><th>Name/Value</th><th>Description</th><th>FbEq</th></tr>
<tr><td></td><td>FBA</td><td>The start and stop commands are taken from the fieldbus Control Word selected by parameter 50.15 Fb cw used.</td><td>3</td></tr>
<tr><td></td><td>D2D</td><td>Reserved.</td><td>4</td></tr>
<tr><td></td><td>In1F In2R</td><td>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.<br>
<table>
<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>
<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>
</table></td><td>5</td></tr>
<tr><td></td><td>In1St In2Dir</td><td>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).</td><td>6</td></tr>
<tr><td>10.05</td><td>Ext2 start in1</td><td>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.<br>Note: This parameter cannot be changed while the drive is running.</td><td></td></tr>
<tr><td></td><td>DI1</td><td>Digital input DI1 (as indicated by 02.01 DI status, bit 0).</td><td>1073742337</td></tr>
<tr><td></td><td>DI2</td><td>Digital input DI2 (as indicated by 02.01 DI status, bit 1).</td><td>1073807873</td></tr>
<tr><td></td><td>DIO4</td><td>Digital input/output DIO4 (as indicated by 02.03 DIO status, bit 3).</td><td>1073938947</td></tr>
<tr><td></td><td>Timed func</td><td>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.</td><td>1074005518</td></tr>
<tr><td></td><td>Const</td><td rowspan="2">Bit pointer setting (see Terms and abbreviations on page 109).</td><td rowspan="2">-</td></tr>
<tr><td></td><td>Pointer</td></tr>
<tr><td>10.06</td><td>Ext2 start in2</td><td>Selects source 2 of start and stop commands for external control location EXT2. See parameter 10.04 Ext2 start func, selection 3-wire.<br>Note: This parameter cannot be changed while the drive is running.</td><td></td></tr>
<tr><td></td><td>DI2</td><td>Digital input DI2 (as indicated by 02.01 DI status, bit 1).</td><td>1073807873</td></tr>
<tr><td></td><td>DI5</td><td>Digital input DI5 (as indicated by 02.01 DI status, bit 4).</td><td>1074004481</td></tr>
<tr><td></td><td>Const</td><td rowspan="2">Bit pointer setting (see Terms and abbreviations on page 109).</td><td rowspan="2">-</td></tr>
<tr><td></td><td>Pointer</td></tr>
<tr><td>10.10</td><td>Fault reset sel</td><td>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.<br>0 -> 1 = Fault reset.</td><td></td></tr>
<tr><td></td><td>DI1</td><td>Digital input DI1 (as indicated by 02.01 DI status, bit 0).</td><td>1073742337</td></tr>
<tr><td></td><td>DI2</td><td>Digital input DI2 (as indicated by 02.01 DI status, bit 1).</td><td>1073807873</td></tr>
<tr><td></td><td>DI3</td><td>Digital input DI3 (as indicated by 02.01 DI status, bit 2).</td><td>1073873409</td></tr>
<tr><td></td><td>DI4</td><td>Digital input DI4 (as indicated by 02.01 DI status, bit 3).</td><td>1073938945</td></tr>
<tr><td></td><td>DI5</td><td>Digital input DI5 (as indicated by 02.01 DI status, bit 4).</td><td>1074004481</td></tr>
</table>

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 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.<br>1 = Run enable.<br>**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 |
| | 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 *109*). | - |
| | 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*.<br>0 = OFF3 active.<br>**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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 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*).<br>0 = OFF1 active.<br>**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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |

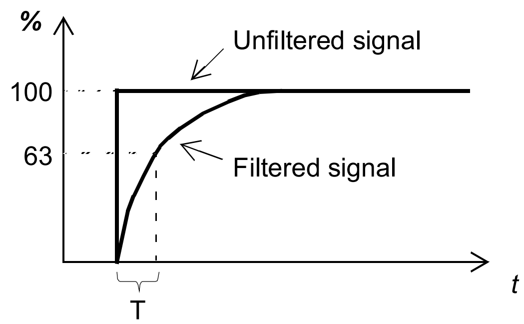
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 10.17 | Start enable | Selects the source for the Start enable signal.<br>1 = Start enable.<br>If the signal is switched off, the drive will not start or coasts to stop if running. | |
| | DI1 | Digital input DI1 (as indicated by *02.01 DI 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 10.19 | Start inhibit | Enables the start inhibit function. The function prevents drive restart (i.e. protects against unexpected start) if<br>• the drive trips on a fault and the fault is reset,<br>• the run enable signal is activated while the start command is active (see parameter *10.11 Run enable*),<br>• control changes from local to remote, or<br>• external control switches from EXT1 to EXT2 or vice versa.<br>A new rising edge of the start command is needed after the start inhibit has been activated.<br>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 intrl func | Defines how the start interlock input (DIIL) on the JCU control unit affects the drive operation. | |
| | Off2 stop | With the drive running:<br>• 1 = Normal operation.<br>• 0 = Stop by coasting. The drive can be restarted by restoring the start interlock signal and switching the start signal from 0 to 1.<br>With the drive stopped:<br>• 1 = Starting allowed.<br>• 0 = Starting not allowed. | 0 |
| | Off3 stop | With the drive running:<br>• 1 = Normal operation.<br>• 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.<br>With the drive stopped:<br>• 1 = Starting allowed.<br>• 0 = Starting not allowed. | 1 |

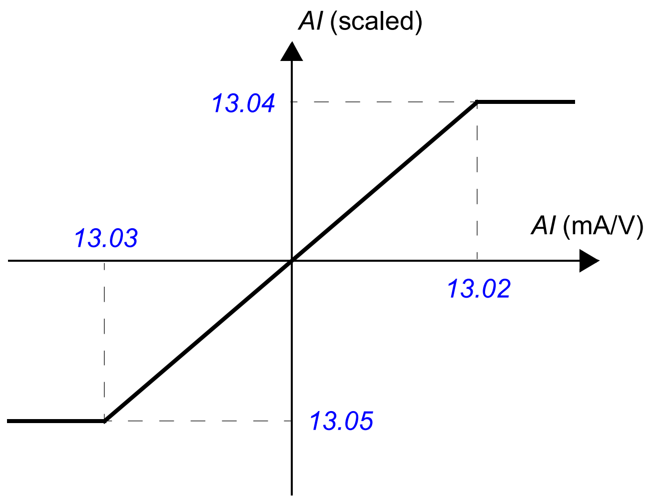
<table>
<tr><th>No.</th><th>Name/Value</th><th>Description</th><th>FbEq</th></tr>
<tr><td colspan="2">11 Start/stop mode</td><td>Start and stop modes; magnetization settings; DC hold configuration.</td><td></td></tr>
<tr><td>11.01</td><td>Start mode</td><td>Selects the motor start function.<br>Notes:<br>• Selections Fast and Const time are ignored if parameter 99.05 is set to Scalar.<br>• Starting to a rotating machine is not possible when DC magnetizing is selected (Fast or Const time).</td><td></td></tr>
<tr><td></td><td>Fast</td><td>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.<br>Note: This parameter cannot be changed while the drive is running.</td><td>0</td></tr>
<tr><td></td><td>Const time</td><td>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.<br>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.</td><td>1</td></tr>
<tr><td></td><td>Automatic</td><td>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.<br>Note: If parameter 99.05 Motor ctrl mode is set to Scalar, no flying start or automatic restart is possible by default.</td><td>2</td></tr>
<tr><td>11.02</td><td>Dc-magn time</td><td>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.<br>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:<br><table><tr><th>Motor rated power</th><th>Constant magnetizing time</th></tr><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></table>Note: This parameter cannot be changed while the drive is running.</td><td></td></tr>
<tr><td></td><td>0 … 10000 ms</td><td>Constant DC magnetizing time.</td><td>1 = 1 ms</td></tr>
</table>

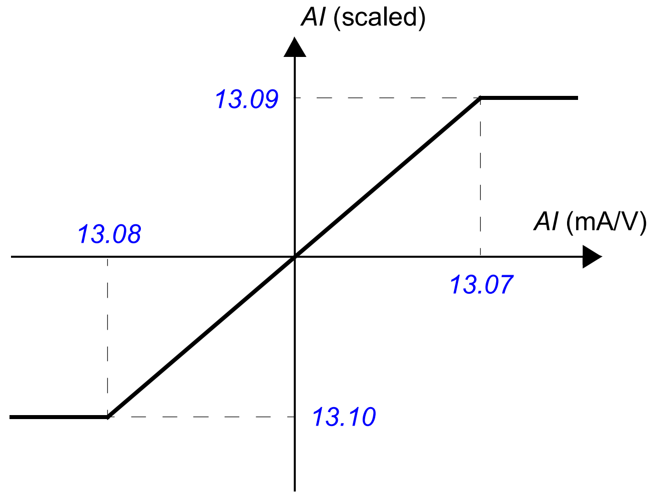
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 11.03 | Stop mode | Selects the motor stop function. | |
| | Coast | Stop by cutting of the motor power supply. The motor coasts to a stop.<br>**WARNING!** If the mechanical brake is used, ensure it is safe to stop the drive by coasting. | 1 |
| | Ramp | Stop along ramp. See parameter group *22 Speed ref ramp* on page *173*. | 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% |
| 11.06 | Dc hold | Enables the DC hold function. The function makes it possible to lock the rotor at zero speed.<br>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.<br><br>**Notes:**<br>• The DC hold function has no effect if the start signal is switched off.<br>• The DC hold function can only be activated in speed control mode.<br>• The DC hold function cannot be activated if parameter *99.05 Motor ctrl mode* is set to *Scalar*.<br>• 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 |

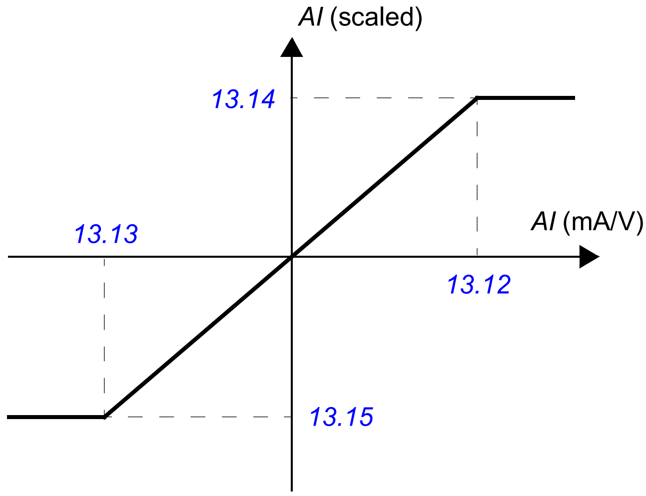
| | | | |
|---|---|---|---|
| **12 Operating mode** | | Selection of external control location and EXT2 operating mode. | |
| 12.01 | Ext1/Ext2 sel | Selects the external control location (EXT1 or EXT2), or the source of a selection signal (0 = EXT1; 1 = EXT2). | |
| | Ext1 | EXT1 is active. | 0 |

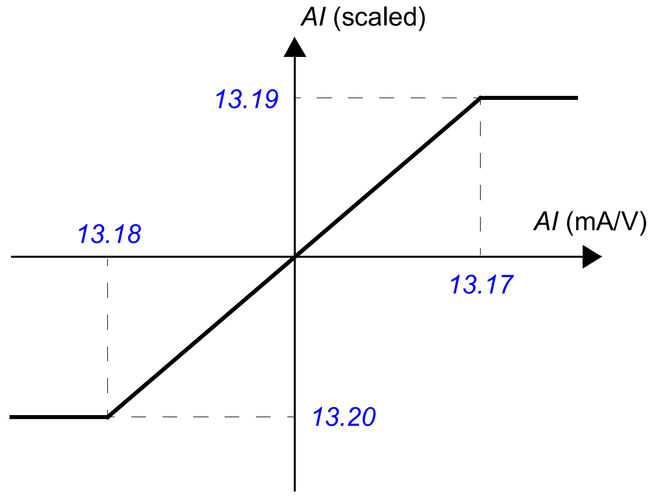
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Ext2 | EXT2 is active. | 1 |
| | DI1 | The external control location is determined by the status of digital input DI1 (as indicated by *02.01 DI status*, bit 0). | 1073742337 |
| | DI2 | The external control location is determined by the status of digital input DI2 (as indicated by *02.01 DI status*, bit 1). | 1073807873 |
| | DI3 | The external control location is determined by the status of digital input DI3 (as indicated by *02.01 DI status*, bit 2). | 1073873409 |
| | DI4 | The external control location is determined by the status of digital input DI4 (as indicated by *02.01 DI status*, bit 3). | 1073938945 |
| | DI5 | The external control location is determined by the status of digital input DI5 (as indicated by *02.01 DI status*, bit 4). | 1074004481 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 12.05 | Ext2 ctrl mode | Selects the operating mode for external control location EXT2. | |
| | Speed | Speed control. The reference is taken from the source defined by parameter *21.02 Speed ref2 sel*. | 1 |
| | PID | PID control. | 2 |

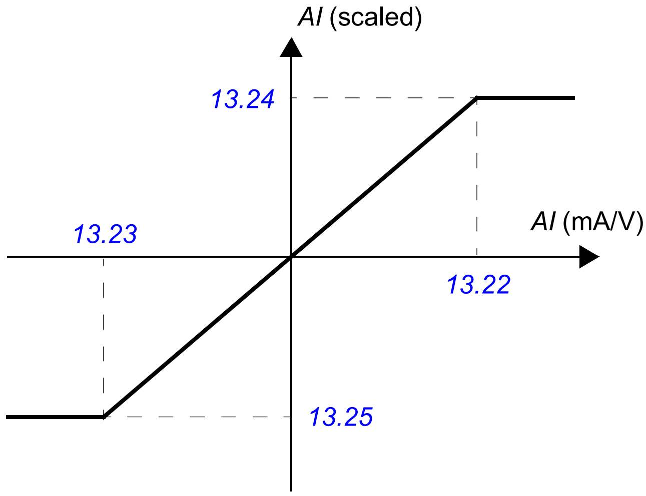
| **13 Analogue inputs** | | Analog input signal processing. | |
|---|---|---|---|
| 13.01 | AI1 filt time | Defines the filter time constant for analog input AI1.<br><br><br>$O = I \times (1 - e^{-t/T})$<br><br>I = filter input (step)<br>O = filter output<br>*t* = time<br>T = filter time constant<br><br>**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. | |
| | 0.000 … 30.000 s | Filter time constant. | 1000 = 1 s |
| 13.02 | AI1 max | Defines the maximum value for analog input AI1. The input type is selected a jumper on the JCU Control Unit (see the *Hardware Manual* of the drive). | |
| | -22.000 … 22.000 mA or<br>-11.000 … 11.000 V | Maximum AI1 value. | 1000 = 1 unit |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 13.03 | AI1 min | Defines the minimum value for analog input AI1. The input type is selected with a jumper on the JCU Control Unit (see the *Hardware Manual* of the drive).. | |
| | -22.000 … 22.000 mA or<br>-11.000 … 11.000 V | Minimum AI1 value. | 1000 = 1 unit |
| 13.04 | AI1 max scale | Defines the real value that corresponds to the maximum analog input AI1 value defined by parameter *13.02 AI1 max*.<br> | |
| | -32768.000 … 32768.000 | Real value corresponding to maximum AI1 value. | 1000 = 1 |
| 13.05 | AI1 min scale | Defines the real value that corresponds to the minimum analog input AI1 value defined by parameter *13.03 AI1 min*. See the drawing at parameter *13.04 AI1 max scale*. | |
| | -32768.000 …32768.000 | Real value corresponding to minimum AI1 value. | 1000 = 1 |
| 13.06 | AI2 filt time | Defines the filter time constant for analog input AI2. See parameter *13.01 AI1 filt time*. | |
| | 0.000 … 30.000 s | Filter time constant. | 1000 = 1 s |
| 13.07 | AI2 max | Defines the maximum value for analog input AI2. The input type is selected with a jumper on the JCU Control Unit (see the *Hardware Manual* of the drive).. | |
| | -22.000 … 22.000 mA or<br>-11.000 … 11.000 V | AI2 maximum value. | 1000 = 1 unit |
| 13.08 | AI2 min | Defines the minimum value for analog input AI2. The input type is selected with a jumper on the JCU Control Unit (see the *Hardware Manual* of the drive).. | |
| | -22.000 … 22.000 mA or<br>-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 analog input AI2 value defined by parameter *13.07 AI2 max*.<br><br> | |
| | -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 analog 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 analog 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 analog 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 analog 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 | Defines the real value that corresponds to the maximum analog input AI3 value defined by parameter *13.12 AI3 max*.<br><br> | |
| | -32768.000 … 32768.000 | Real value corresponding to maximum AI3 value. | 1000 = 1 |
| 13.15 | AI3 min scale | Defines the real value that corresponds to the minimum analog input AI3 value defined by parameter *13.13 AI3 min*. See the drawing at parameter *13.14 AI3 max scale*. | |
| | -32768.000 … 32768.000 | Real value corresponding to minimum AI3 value. | 1000 = 1 |
| 13.16 | AI4 filt time | Defines the filter time constant for analog input AI4. See parameter *13.01 AI1 filt time*. | |
| | 0.000 … 30.000 s | Filter time constant. | 1000 = 1 s |
| 13.17 | AI4 max | Defines the maximum value for analog 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. | |
| | -22.000 … 22.000 mA or -11.000 … 11.000 V | AI4 maximum value. | 1000 = 1 unit |
| 13.18 | AI4 min | Defines the minimum value for analog 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. | |
| | -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 analog input AI4 value defined by parameter *13.17 AI4 max*.<br><br> | |
| | -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 analog 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 analog 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 analog 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 analog 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 | Defines the real value that corresponds to the maximum analog input AI5 value defined by parameter *13.22 AI5 max*.<br><br> | |
| | -32768.000 … 32768.000 | Real value corresponding to maximum AI5 value. | 1000 = 1 |
| 13.25 | AI5 min scale | Defines the real value that corresponds to the minimum analog input AI5 value defined by parameter *13.23 AI5 min*.<br>See the drawing at parameter *13.24 AI5 max scale*. | |
| | -32768.000 … 32768.000 | Real value corresponding to minimum AI5 value. | 1000 = 1 |
| 13.31 | AI tune | Triggers the AI tuning function.<br>Connect the signal to the input and select the appropriate tuning function. | |
| | No action | AI tune is not activated. | 0 |
| | AI1 min tune | Current analog input AI1 signal value is set as minimum value of AI1 into parameter *13.03 AI1 min*. The value reverts back to *No action* automatically. | 1 |
| | AI1 max tune | Current analog input AI1 signal value is set as maximum value of AI1 into parameter *13.02 AI1 max*. The value reverts back to *No action* automatically. | 2 |
| | AI2 min tune | Current analog input AI2 signal value is set as minimum value of AI2 into parameter *13.08 AI2 min*. The value reverts back to *No action* automatically. | 3 |
| | AI2 max tune | Current analog input AI2 signal value is set as maximum value of AI2 into parameter *13.07 AI2 max*. The value reverts back to *No action* automatically. | 4 |
| 13.32 | AI superv func | Selects how the drive reacts when analog 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 (0x8110)* fault. | 1 |
| | Spd ref Safe | The drive generates an *AI SUPERVISION (0x8110)* alarm and sets the speed to the speed defined by parameter *30.02 Speed ref safe*.<br>**WARNING!** Make sure that it is safe to continue operation in case of a communication break. | 2 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Last speed | The drive generates an *AI SUPERVISION (0x8110)* 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.<br>**WARNING!** Make sure that it is safe to continue operation in case of a communication break. | 3 |
| 13.33 | AI superv cw | Selects the analog input signal supervision limit. | |

| Bit | Supervision | Action selected by parameter *13.32 AI superv func* is taken if |
|---|---|---|
| 0 | AI1 min sup | AI1 signal value falls below the value defined by equation: par. *13.03 AI1 min* - 0.5 mA or V |
| 1 | AI1 max sup | AI1 signal value exceeds the value defined by equation: par. *13.02 AI1 max* + 0.5 mA or V |
| 2 | AI2 min sup | AI2 signal value falls below the value defined by equation: par. *13.08 AI2 min* - 0.5 mA or V |
| 3 | AI2 max sup | AI1 signal value exceeds the value defined by equation: par. *13.07 AI2 max* + 0.5 mA or V |

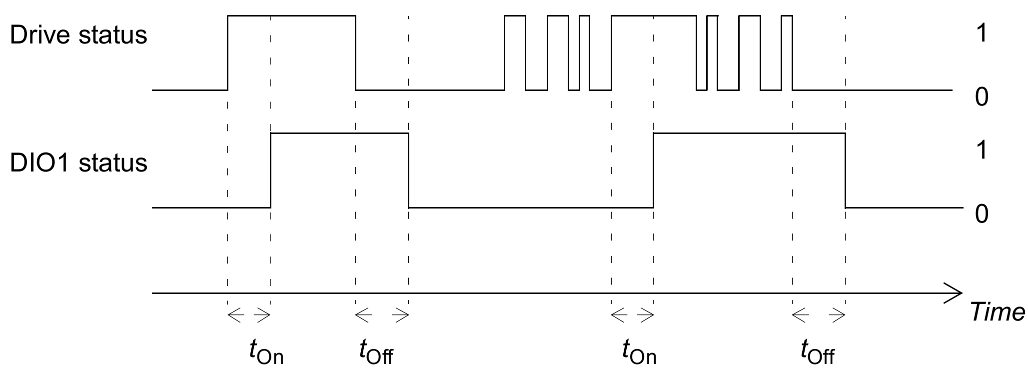
**Example:** If parameter value is set to 0b0010, bit 1 *AI1 max sup* is selected.

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| **14 Digital I/O** | | Configuration of digital input/outputs, relay outputs, the frequency input, and the frequency output. | |
| 14.01 | DI invert mask | Inverts status of digital inputs as reported by *02.01 DI status*. | |

| Bit | Name |
|---|---|
| 0 | 1 = Invert DI1 |
| 1 | 1 = Invert DI2 |
| 2 | 1 = Invert DI3 |
| 3 | 1 = Invert DI4 |
| 4 | 1 = Invert DI5 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 14.02 | DIO1 conf | Selects whether DIO1 is used as a digital output or 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 *Output*). | |
| | Trad pump7 | Bit 6 of *05.02 Trad pump cmd* (see page *125*). | 1074136322 |
| | Trad pump8 | Bit 7 of *05.02 Trad pump cmd* (see page *125*). | 1074201858 |
| | Ready | Bit 0 of *06.01 Status word1* (see page *126*). | 1073743361 |
| | Enabled | Bit 1 of *06.01 Status word1* (see page *126*). | 1073808897 |
| | Started | Bit 2 of *06.01 Status word1* (see page *126*). | 1073874433 |
| | Running | Bit 3 of *06.01 Status word1* (see page *126*). | 1073939969 |
| | Alarm | Bit 7 of *06.01 Status word1* (see page *126*). | 1074202113 |
| | Ext2 active | Bit 8 of *06.01 Status word1* (see page *126*). | 1074267649 |
| | Fault | Bit 10 of *06.01 Status word1* (see page *126*). | 1074398721 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Fault(-1) | Bit 12 of *06.01 Status word1* (see page *126*). | 1074529793 |
| | Ready relay | Bit 2 of *06.02 Status word2* (see page *127*). | 1073874434 |
| | RunningRelay | Bit 3 of *06.02 Status word2* (see page *127*). | 1073939970 |
| | Ref running | Bit 4 of *06.02 Status word2* (see page *127*). | 1074005506 |
| | Charge ready | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Neg speed | Bit 0 of *06.03 Speed ctrl stat* (see page *128*). | 1073743363 |
| | Zero speed | Bit 1 of *06.03 Speed ctrl stat* (see page *128*). | 1073808899 |
| | Above limit | Bit 2 of *06.03 Speed ctrl stat* (see page *128*). | 1073874435 |
| | At setpoint | Bit 3 of *06.03 Speed ctrl stat* (see page *128*). | 1073939971 |
| | Supervision1 | Bit 0 of *06.13 Superv status* (see page *129*). | 1073743373 |
| | Supervision2 | Bit 1 of *06.13 Superv status* (see page *129*). | 1073808909 |
| | Supervision3 | Bit 2 of *06.13 Superv status* (see page *129*). | 1073874445 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 14.04 | DIO1 Ton | Defines the on (activation) delay for digital input/output DIO1 when *14.02 DIO1 conf* is set to *Output*. | |

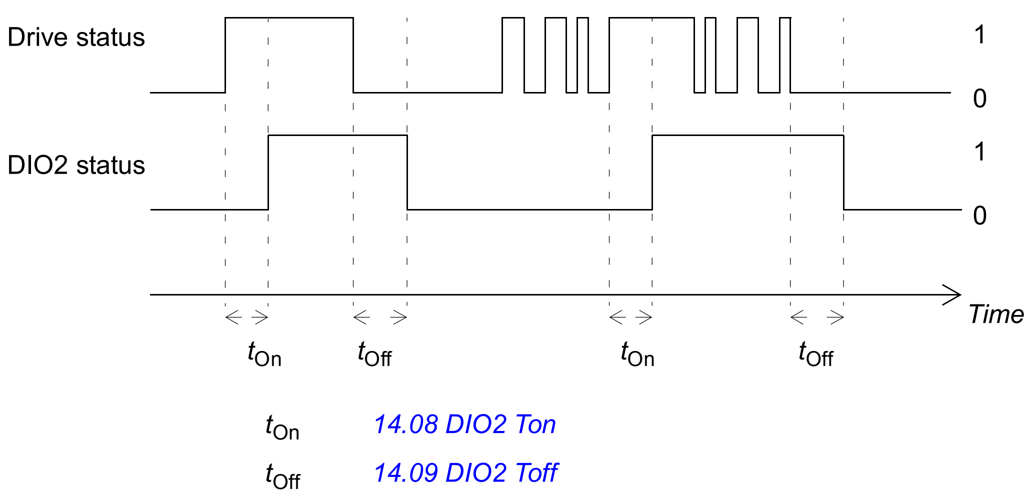



$t_{On}$ *14.04 DIO1 Ton*

$t_{Off}$ *14.05 DIO1 Toff*

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 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 *14.02 DIO1 conf* is set to *Output*. See parameter *14.04 DIO1 Ton*. | |
| | 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 input. | |
| | Output | DIO2 is used as a digital output. | 0 |
| | Input | DIO2 is used as a digital input. | 1 |
| | Freq output | DIO2 is used as a frequency output. | 2 |
| 14.07 | DIO2 out src | Selects a drive signal to be connected to digital output DIO2 (when *14.06 DIO2 conf* is set to *Output*). | |
| | Ready | Bit 0 of *06.01 Status word1* (see page *126*). | 1073743361 |
| | Enabled | Bit 1 of *06.01 Status word1* (see page *126*). | 1073808897 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Started | Bit 2 of *06.01 Status word1* (see page *126*). | 1073874433 |
| | Running | Bit 3 of *06.01 Status word1* (see page *126*). | 1073939969 |
| | Alarm | Bit 7 of *06.01 Status word1* (see page *126*). | 1074202113 |
| | Ext2 active | Bit 8 of *06.01 Status word1* (see page *126*). | 1074267649 |
| | Fault | Bit 10 of *06.01 Status word1* (see page *126*). | 1074398721 |
| | Fault(-1) | Bit 12 of *06.01 Status word1* (see page *126*). | 1074529793 |
| | Ready relay | Bit 2 of *06.02 Status word2* (see page *127*). | 1073874434 |
| | RunningRelay | Bit 3 of *06.02 Status word2* (see page *127*). | 1073939970 |
| | Ref running | Bit 4 of *06.02 Status word2* (see page *127*). | 1074005506 |
| | Charge ready | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Neg speed | Bit 0 of *06.03 Speed ctrl stat* (see page *128*). | 1073743363 |
| | Zero speed | Bit 1 of *06.03 Speed ctrl stat* (see page *128*). | 1073808899 |
| | Above limit | Bit 2 of *06.03 Speed ctrl stat* (see page *128*). | 1073874435 |
| | At setpoint | Bit 3 of *06.03 Speed ctrl stat* (see page *128*). | 1073939971 |
| | Supervision1 | Bit 0 of *06.13 Superv status* (see page *129*). | 1073743373 |
| | Supervision2 | Bit 1 of *06.13 Superv status* (see page *129*). | 1073808909 |
| | Supervision3 | Bit 2 of *06.13 Superv status* (see page *129*). | 1073874445 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 14.08 | DIO2 Ton | Defines the on (activation) delay for digital input/output DIO2 when *14.06 DIO2 conf* is set to *Output*. | |

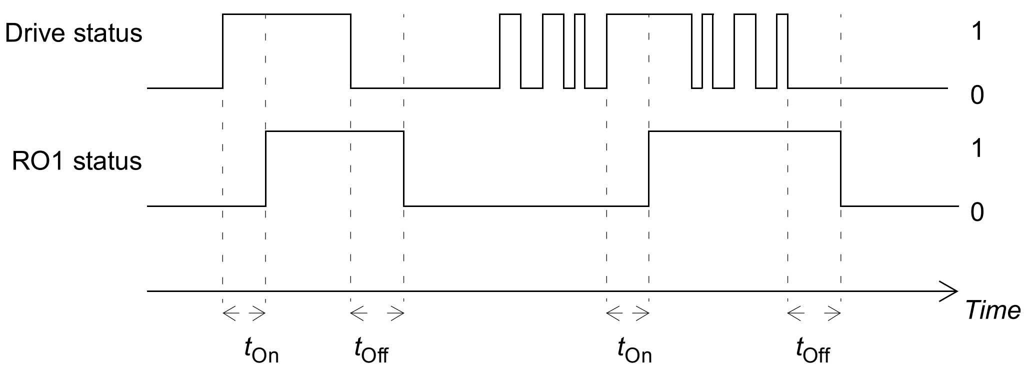



| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 0.0 … 3000.0 s | On (activation) delay for DIO2 when set as an output. | 10 = 1 s |
| 14.09 | DIO2 Toff | Defines the off (deactivation) delay for digital input/output DIO2 when *14.06 DIO2 conf* is set to *Output*. See parameter *14.08 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 digital input. | |
| | Output | DIO3 is used as a digital output. | 0 |
| | Input | DIO3 is used as a digital input. | 1 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 14.11 | DIO3 out src | Selects a drive signal to be connected to digital output DIO3 (when *14.10 DIO3 conf* is set to *Output*). | |
| | Ready | Bit 0 of *06.01 Status word1* (see page *126*). | 1073743361 |
| | Enabled | Bit 1 of *06.01 Status word1* (see page *126*). | 1073808897 |
| | Started | Bit 2 of *06.01 Status word1* (see page *126*). | 1073874433 |
| | Running | Bit 3 of *06.01 Status word1* (see page *126*). | 1073939969 |
| | Alarm | Bit 7 of *06.01 Status word1* (see page *126*). | 1074202113 |
| | Ext2 active | Bit 8 of *06.01 Status word1* (see page *126*). | 1074267649 |
| | Fault | Bit 10 of *06.01 Status word1* (see page *126*). | 1074398721 |
| | Fault(-1) | Bit 12 of *06.01 Status word1* (see page *126*). | 1074529793 |
| | Ready relay | Bit 2 of *06.02 Status word2* (see page *127*). | 1073874434 |
| | RunningRelay | Bit 3 of *06.02 Status word2* (see page *127*). | 1073939970 |
| | Ref running | Bit 4 of *06.02 Status word2* (see page *127*). | 1074005506 |
| | Charge ready | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Neg speed | Bit 0 of *06.03 Speed ctrl stat* (see page *128*). | 1073743363 |
| | Zero speed | Bit 1 of *06.03 Speed ctrl stat* (see page *128*). | 1073808899 |
| | Above limit | Bit 2 of *06.03 Speed ctrl stat* (see page *128*). | 1073874435 |
| | At setpoint | Bit 3 of *06.03 Speed ctrl stat* (see page *128*). | 1073939971 |
| | Supervision1 | Bit 0 of *06.13 Superv status* (see page *129*). | 1073743373 |
| | Supervision2 | Bit 1 of *06.13 Superv status* (see page *129*). | 1073808909 |
| | Supervision3 | Bit 2 of *06.13 Superv status* (see page *129*). | 1073874445 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | 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 *Output*). | |
| | Ready | Bit 0 of *06.01 Status word1* (see page *126*). | 1073743361 |
| | Enabled | Bit 1 of *06.01 Status word1* (see page *126*). | 1073808897 |
| | Started | Bit 2 of *06.01 Status word1* (see page *126*). | 1073874433 |
| | Running | Bit 3 of *06.01 Status word1* (see page *126*). | 1073939969 |
| | Alarm | Bit 7 of *06.01 Status word1* (see page *126*). | 1074202113 |
| | Ext2 active | Bit 8 of *06.01 Status word1* (see page *126*). | 1074267649 |
| | Fault | Bit 10 of *06.01 Status word1* (see page *126*). | 1074398721 |
| | Fault(-1) | Bit 12 of *06.01 Status word1* (see page *126*). | 1074529793 |
| | Ready relay | Bit 2 of *06.02 Status word2* (see page *127*). | 1073874434 |
| | RunningRelay | Bit 3 of *06.02 Status word2* (see page *127*). | 1073939970 |
| | Ref running | Bit 4 of *06.02 Status word2* (see page *127*). | 1074005506 |
| | Charge ready | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Neg speed | Bit 0 of *06.03 Speed ctrl stat* (see page *128*). | 1073743363 |
| | Zero speed | Bit 1 of *06.03 Speed ctrl stat* (see page *128*). | 1073808899 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Above limit | Bit 2 of *06.03 Speed ctrl stat* (see page *128*). | 1073874435 |
| | At setpoint | Bit 3 of *06.03 Speed ctrl stat* (see page *128*). | 1073939971 |
| | Supervision1 | Bit 0 of *06.13 Superv status* (see page *129*). | 1073743373 |
| | Supervision2 | Bit 1 of *06.13 Superv status* (see page *129*). | 1073808909 |
| | Supervision3 | Bit 2 of *06.13 Superv status* (see page *129*). | 1073874445 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 14.42 | RO1 src | Selects a drive signal to be connected to relay output RO1. | |
| | Trad pump1 | Bit 0 of *05.02 Trad pump cmd* (see page *125*). | 1073743106 |
| | Ready | Bit 0 of *06.01 Status word1* (see page *126*). | 1073743361 |
| | Enabled | Bit 1 of *06.01 Status word1* (see page *126*). | 1073808897 |
| | Started | Bit 2 of *06.01 Status word1* (see page *126*). | 1073874433 |
| | Running | Bit 3 of *06.01 Status word1* (see page *126*). | 1073939969 |
| | Alarm | Bit 7 of *06.01 Status word1* (see page *126*). | 1074202113 |
| | Ext2 active | Bit 8 of *06.01 Status word1* (see page *126*). | 1074267649 |
| | Fault | Bit 10 of *06.01 Status word1* (see page *126*). | 1074398721 |
| | Fault(-1) | Bit 12 of *06.01 Status word1* (see page *126*). | 1074529793 |
| | Ready relay | Bit 2 of *06.02 Status word2* (see page *127*). | 1073874434 |
| | RunningRelay | Bit 3 of *06.02 Status word2* (see page *127*). | 1073939970 |
| | Ref running | Bit 4 of *06.02 Status word2* (see page *127*). | 1074005506 |
| | Charge ready | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Neg speed | Bit 0 of *06.03 Speed ctrl stat* (see page *128*). | 1073743363 |
| | Zero speed | Bit 1 of *06.03 Speed ctrl stat* (see page *128*). | 1073808899 |
| | Above limit | Bit 2 of *06.03 Speed ctrl stat* (see page *128*). | 1073874435 |
| | At setpoint | Bit 3 of *06.03 Speed ctrl stat* (see page *128*). | 1073939971 |
| | Supervision1 | Bit 0 of *06.13 Superv status* (see page *129*). | 1073743373 |
| | Supervision2 | Bit 1 of *06.13 Superv status* (see page *129*). | 1073808909 |
| | Supervision3 | Bit 2 of *06.13 Superv status* (see page *129*). | 1073874445 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 14.43 | RO1 Ton | Defines the on (activation) delay for relay output RO1. | |



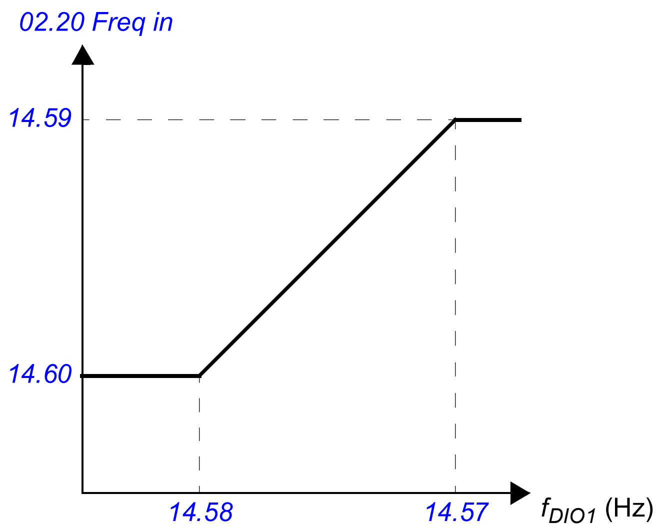

$t_{On}$ *14.43 RO1 Ton*

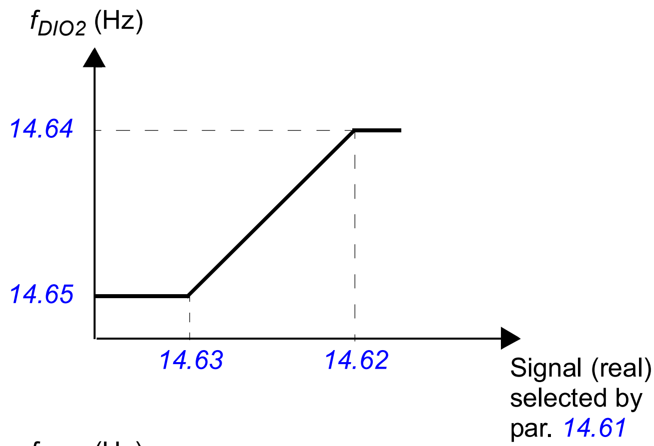
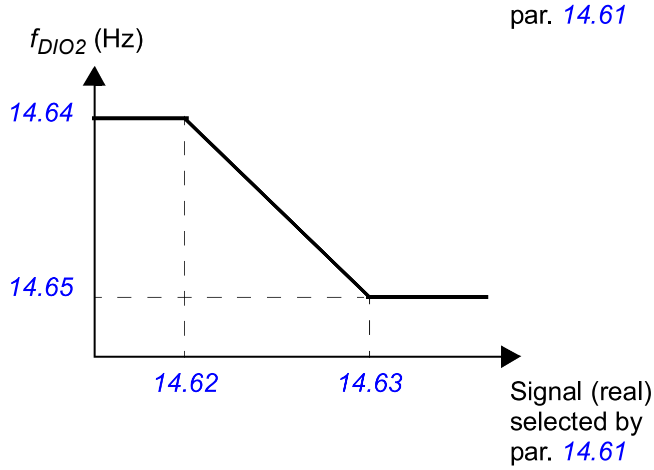
$t_{Off}$ *14.44 RO1 Toff*

| | | | |
|---|---|---|---|
| | 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. | |
| | Trad pump2 | Bit 1 of *05.02 Trad pump cmd* (see page *125*). | 1073808642 |
| | Ready | Bit 0 of *06.01 Status word1* (see page *126*). | 1073743361 |
| | Enabled | Bit 1 of *06.01 Status word1* (see page *126*). | 1073808897 |
| | Started | Bit 2 of *06.01 Status word1* (see page *126*). | 1073874433 |
| | Running | Bit 3 of *06.01 Status word1* (see page *126*). | 1073939969 |
| | Alarm | Bit 7 of *06.01 Status word1* (see page *126*). | 1074202113 |
| | Ext2 active | Bit 8 of *06.01 Status word1* (see page *126*). | 1074267649 |
| | Fault | Bit 10 of *06.01 Status word1* (see page *126*). | 1074398721 |
| | Fault(-1) | Bit 12 of *06.01 Status word1* (see page *126*). | 1074529793 |
| | Ready relay | Bit 2 of *06.02 Status word2* (see page *127*). | 1073874434 |
| | RunningRelay | Bit 3 of *06.02 Status word2* (see page *127*). | 1073939970 |
| | Ref running | Bit 4 of *06.02 Status word2* (see page *127*). | 1074005506 |
| | Charge ready | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Neg speed | Bit 0 of *06.03 Speed ctrl stat* (see page *128*). | 1073743363 |
| | Zero speed | Bit 1 of *06.03 Speed ctrl stat* (see page *128*). | 1073808899 |
| | Above limit | Bit 2 of *06.03 Speed ctrl stat* (see page *128*). | 1073874435 |
| | At setpoint | Bit 3 of *06.03 Speed ctrl stat* (see page *128*). | 1073939971 |
| | Supervision1 | Bit 0 of *06.13 Superv status* (see page *129*). | 1073743373 |
| | Supervision2 | Bit 1 of *06.13 Superv status* (see page *129*). | 1073808909 |
| | Supervision3 | Bit 2 of *06.13 Superv status* (see page *129*). | 1073874445 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 14.48 | RO3 src | Selects a drive signal to be connected to relay output RO3. | |
| | Trad pump3 | Bit 2 of *05.02 Trad pump cmd* (see page *125*). | 1073874178 |
| | Ready | Bit 0 of *06.01 Status word1* (see page *126*). | 1073743361 |
| | Enabled | Bit 1 of *06.01 Status word1* (see page *126*). | 1073808897 |
| | Started | Bit 2 of *06.01 Status word1* (see page *126*). | 1073874433 |
| | Running | Bit 3 of *06.01 Status word1* (see page *126*). | 1073939969 |
| | Alarm | Bit 7 of *06.01 Status word1* (see page *126*). | 1074202113 |
| | Ext2 active | Bit 8 of *06.01 Status word1* (see page *126*). | 1074267649 |
| | Fault | Bit 10 of *06.01 Status word1* (see page *126*). | 1074398721 |
| | Fault(-1) | Bit 12 of *06.01 Status word1* (see page *126*). | 1074529793 |
| | Ready relay | Bit 2 of *06.02 Status word2* (see page *127*). | 1073874434 |
| | RunningRelay | Bit 3 of *06.02 Status word2* (see page *127*). | 1073939970 |
| | Ref running | Bit 4 of *06.02 Status word2* (see page *127*). | 1074005506 |
| | Charge ready | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Neg speed | Bit 0 of *06.03 Speed ctrl stat* (see page *128*). | 1073743363 |
| | Zero speed | Bit 1 of *06.03 Speed ctrl stat* (see page *128*). | 1073808899 |
| | Above limit | Bit 2 of *06.03 Speed ctrl stat* (see page *128*). | 1073874435 |
| | At setpoint | Bit 3 of *06.03 Speed ctrl stat* (see page *128*). | 1073939971 |
| | Supervision1 | Bit 0 of *06.13 Superv status* (see page *129*). | 1073743373 |
| | Supervision2 | Bit 1 of *06.13 Superv status* (see page *129*). | 1073808909 |
| | Supervision3 | Bit 2 of *06.13 Superv status* (see page *129*). | 1073874445 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 14.51 | RO4 src | Selects a drive signal to be connected to relay output RO4. | |
| | Trad pump4 | Bit 3 of *05.02 Trad pump cmd* (see page *125*). | 1073939714 |
| | Ready | Bit 0 of *06.01 Status word1* (see page *126*). | 1073743361 |
| | Enabled | Bit 1 of *06.01 Status word1* (see page *126*). | 1073808897 |
| | Started | Bit 2 of *06.01 Status word1* (see page *126*). | 1073874433 |
| | Running | Bit 3 of *06.01 Status word1* (see page *126*). | 1073939969 |
| | Alarm | Bit 7 of *06.01 Status word1* (see page *126*). | 1074202113 |
| | Ext2 active | Bit 8 of *06.01 Status word1* (see page *126*). | 1074267649 |
| | Fault | Bit 10 of *06.01 Status word1* (see page *126*). | 1074398721 |
| | Fault(-1) | Bit 12 of *06.01 Status word1* (see page *126*). | 1074529793 |
| | Ready relay | Bit 2 of *06.02 Status word2* (see page *127*). | 1073874434 |
| | RunningRelay | Bit 3 of *06.02 Status word2* (see page *127*). | 1073939970 |
| | Ref running | Bit 4 of *06.02 Status word2* (see page *127*). | 1074005506 |
| | Charge ready | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Neg speed | Bit 0 of *06.03 Speed ctrl stat* (see page *128*). | 1073743363 |
| | Zero speed | Bit 1 of *06.03 Speed ctrl stat* (see page *128*). | 1073808899 |
| | Above limit | Bit 2 of *06.03 Speed ctrl stat* (see page *128*). | 1073874435 |
| | At setpoint | Bit 3 of *06.03 Speed ctrl stat* (see page *128*). | 1073939971 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Supervision1 | Bit 0 of *06.13 Superv status* (see page *129*). | 1073743373 |
| | Supervision2 | Bit 1 of *06.13 Superv status* (see page *129*). | 1073808909 |
| | Supervision3 | Bit 2 of *06.13 Superv status* (see page *129*). | 1073874445 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 14.54 | RO5 src | Selects a drive signal to be connected to relay output RO5. | |
| | Trad pump5 | Bit 4 of *05.02 Trad pump cmd* (see page *125*). | 1074005250 |
| | Ready | Bit 0 of *06.01 Status word1* (see page *126*). | 1073743361 |
| | Enabled | Bit 1 of *06.01 Status word1* (see page *126*). | 1073808897 |
| | Started | Bit 2 of *06.01 Status word1* (see page *126*). | 1073874433 |
| | Running | Bit 3 of *06.01 Status word1* (see page *126*). | 1073939969 |
| | Alarm | Bit 7 of *06.01 Status word1* (see page *126*). | 1074202113 |
| | Ext2 active | Bit 8 of *06.01 Status word1* (see page *126*). | 1074267649 |
| | Fault | Bit 10 of *06.01 Status word1* (see page *126*). | 1074398721 |
| | Fault(-1) | Bit 12 of *06.01 Status word1* (see page *126*). | 1074529793 |
| | Ready relay | Bit 2 of *06.02 Status word2* (see page *127*). | 1073874434 |
| | RunningRelay | Bit 3 of *06.02 Status word2* (see page *127*). | 1073939970 |
| | Ref running | Bit 4 of *06.02 Status word2* (see page *127*). | 1074005506 |
| | Charge ready | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Neg speed | Bit 0 of *06.03 Speed ctrl stat* (see page *128*). | 1073743363 |
| | Zero speed | Bit 1 of *06.03 Speed ctrl stat* (see page *128*). | 1073808899 |
| | Above limit | Bit 2 of *06.03 Speed ctrl stat* (see page *128*). | 1073874435 |
| | At setpoint | Bit 3 of *06.03 Speed ctrl stat* (see page *128*). | 1073939971 |
| | Supervision1 | Bit 0 of *06.13 Superv status* (see page *129*). | 1073743373 |
| | Supervision2 | Bit 1 of *06.13 Superv status* (see page *129*). | 1073808909 |
| | Supervision3 | Bit 2 of *06.13 Superv status* (see page *129*). | 1073874445 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |

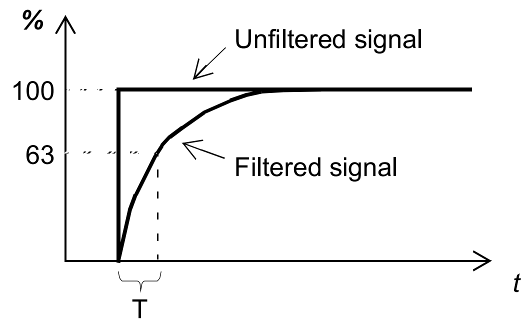
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 14.57 | Freq in max | Defines the maximum input frequency for DIO1 when parameter *14.02 DIO1 conf* is set to *Freq input*.<br>The frequency signal connected to DIO1 is scaled into an internal signal (*02.20 Freq in*) by parameters *14.57*…*14.60* as follows:<br><br> | |
| | 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 *Freq input*. 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 diagram at 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 *Freq output*). | |
| | | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |

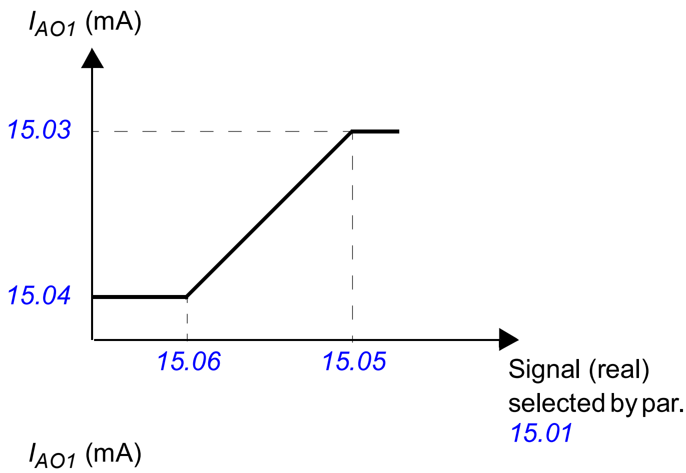
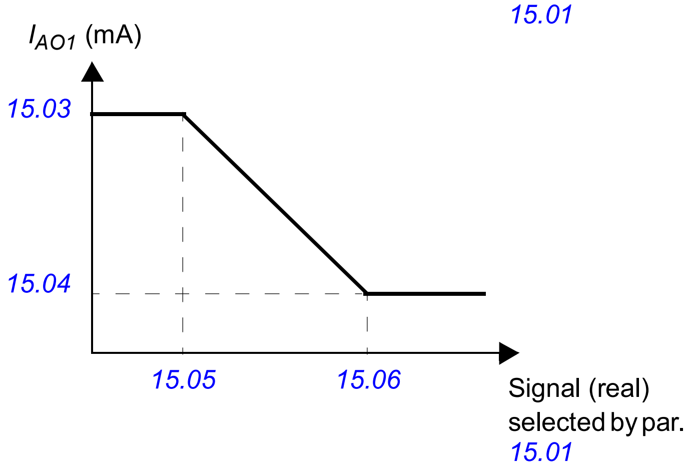
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 14.62 | Freq out max src | When *14.06 DIO2 conf* is set to *Freq output*, defines the real value of the signal (selected by parameter *14.61 Freq out src*) that corresponds to the maximum DIO2 frequency output value (defined by parameter *14.64 Freq out max sca*).<br><br><br><br> | |
| | 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 *Freq output*, 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 *Freq output*, 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 *Freq output*, 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. | |
| | Trad pump6 | Bit 5 of *05.02 Trad pump cmd* (see page *125*). | 1074070786 |
| | Ready | Bit 0 of *06.01 Status word1* (see page *126*). | 1073743361 |
| | Enabled | Bit 1 of *06.01 Status word1* (see page *126*). | 1073808897 |
| | Started | Bit 2 of *06.01 Status word1* (see page *126*). | 1073874433 |

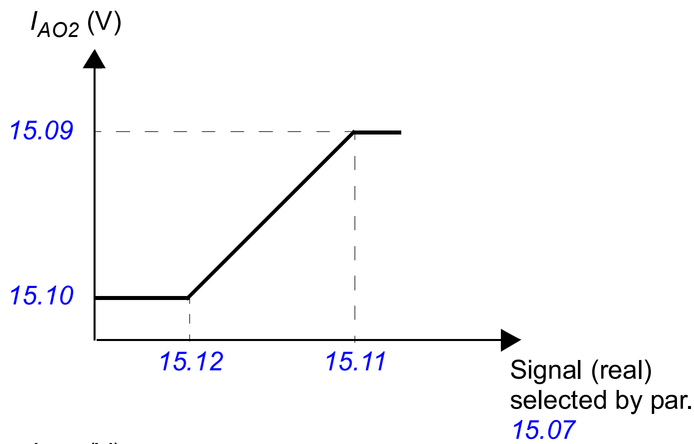
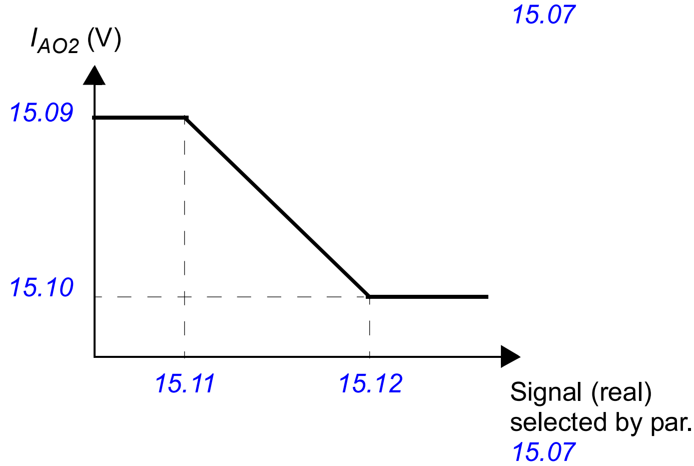
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Running | Bit 3 of *06.01 Status word1* (see page *126*). | 1073939969 |
| | Alarm | Bit 7 of *06.01 Status word1* (see page *126*). | 1074202113 |
| | Ext2 active | Bit 8 of *06.01 Status word1* (see page *126*). | 1074267649 |
| | Fault | Bit 10 of *06.01 Status word1* (see page *126*). | 1074398721 |
| | Fault(-1) | Bit 12 of *06.01 Status word1* (see page *126*). | 1074529793 |
| | Ready relay | Bit 2 of *06.02 Status word2* (see page *127*). | 1073874434 |
| | RunningRelay | Bit 3 of *06.02 Status word2* (see page *127*). | 1073939970 |
| | Ref running | Bit 4 of *06.02 Status word2* (see page *127*). | 1074005506 |
| | Charge ready | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Neg speed | Bit 0 of *06.03 Speed ctrl stat* (see page *128*). | 1073743363 |
| | Zero speed | Bit 1 of *06.03 Speed ctrl stat* (see page *128*). | 1073808899 |
| | Above limit | Bit 2 of *06.03 Speed ctrl stat* (see page *128*). | 1073874435 |
| | At setpoint | Bit 3 of *06.03 Speed ctrl stat* (see page *128*). | 1073939971 |
| | Supervision1 | Bit 0 of *06.13 Superv status* (see page *129*). | 1073743373 |
| | Supervision2 | Bit 1 of *06.13 Superv status* (see page *129*). | 1073808909 |
| | Supervision3 | Bit 2 of *06.13 Superv status* (see page *129*). | 1073874445 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 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) |

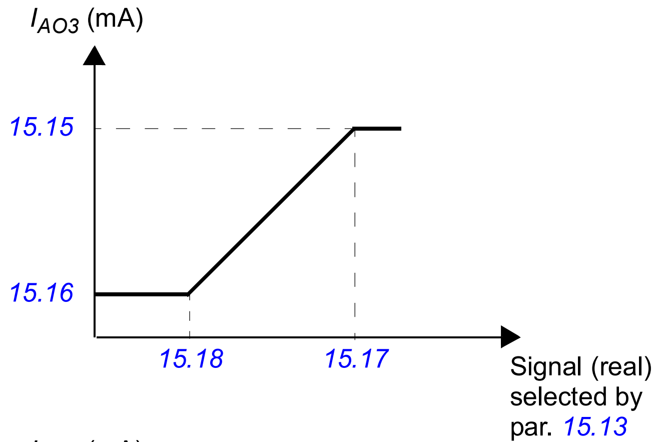
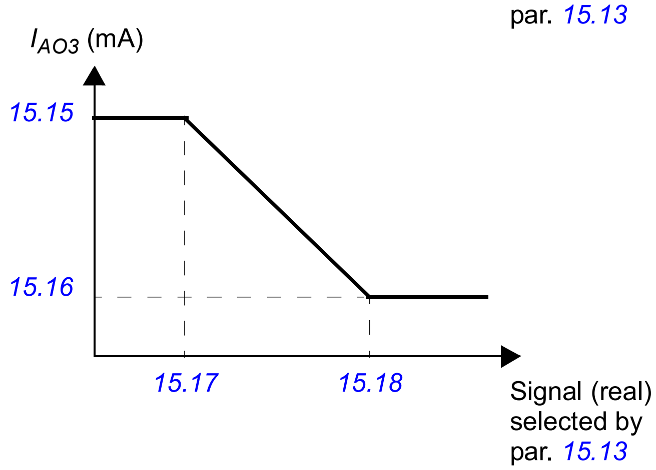
| | | | |
|---|---|---|---|
| **15 Analogue outputs** | | Selection and processing of actual signals to be indicated through the analog outputs.<br>See also section *Programmable analog outputs* on page *65.* | |
| 15.01 | AO1 src | Selects a drive signal to be connected to analog output AO1. | |
| | Speed rpm | *01.01 Motor speed rpm* (see page *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | Frequency | *01.03 Output frequency* (see page *112*). | 1073742083 |
| | Current | *01.04 Motor current* (see page *112*). | 1073742084 |
| | Current % | *01.05 Motor current %* (see page *112*). | 1073742085 |
| | Torque | *01.06 Motor torque* (see page *112*). | 1073742086 |
| | Dc-voltage | *01.07 Dc-voltage* (see page *112*). | 1073742087 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Power inu | *01.22 Power inu out* (see page *112*). | 1073742102 |
| | Power motor | *01.23 Motor power* (see page *112*). | 1073742103 |
| | SpRef unramp | *03.03 SpeedRef unramp* (see page *123*). | 1073742595 |
| | SpRef ramped | *03.05 SpeedRef ramped* (see page *123*). | 1073742597 |
| | SpRef used | *03.06 SpeedRef used* (see page *123*). | 1073742598 |
| | TorqRef used | *03.14 Torq ref used* (see page *123*). | 1073742606 |
| | Proc PID out | *04.05 Process PID out* (see page *123*). | 1073742853 |
| | Process act% | *04.22 Act val %* (see page *124*). | 1073742870 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 15.02 | AO1 filt time | Defines the filtering time constant for analog output AO1.<br><br><br>$O = I \times (1 - e^{-t/T})$<br><br>I = filter input (step)<br>O = filter output<br>*t* = time<br>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 analog 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 analog output AO1. | |
| | 0.000 … 22.700 mA | Minimum AO1 output value. | 1000 = 1 mA |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 15.05 | AO1 src max | 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*).     | |
| | -32768.000 … 32768.000 | Real signal value corresponding to maximum AO1 output value. | 1000 = 1 |
| 15.06 | AO1 src min | 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*. | |
| | -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 analog output AO2. | |
| | Speed rpm | *01.01 Motor speed rpm* (see page *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | Frequency | *01.03 Output frequency* (see page *112*). | 1073742083 |
| | Current | *01.04 Motor current* (see page *112*). | 1073742084 |
| | Current % | *01.05 Motor current %* (see page *112*). | 1073742085 |
| | Torque | *01.06 Motor torque* (see page *112*). | 1073742086 |
| | Dc-voltage | *01.07 Dc-voltage* (see page *112*). | 1073742087 |
| | Power inu | *01.22 Power inu out* (see page *112*). | 1073742102 |
| | Power motor | *01.23 Motor power* (see page *112*). | 1073742103 |
| | SpRef unramp | *03.03 SpeedRef unramp* (see page *123*). | 1073742595 |
| | SpRef ramped | *03.05 SpeedRef ramped* (see page *123*). | 1073742597 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | SpRef used | *03.06 SpeedRef used* (see page *123*). | 1073742598 |
| | TorqRef used | *03.14 Torq ref used* (see page *123*). | 1073742606 |
| | Proc PID out | *04.05 Process PID out* (see page *123*). | 1073742853 |
| | Process act% | *04.22 Act val %* (see page *124*). | 1073742870 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 15.08 | AO2 filt time | Defines the filtering time constant for analog 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 analog output AO2. | |
| | -10.000 … 10.000 V | Maximum AO2 output value. | 1000 = 1 V |
| 15.10 | AO2 out min | Defines the minimum output value for analog output AO2. | |
| | -10.000 … 10.000 V | Minimum AO2 output value. | 1000 = 1 mA |
| 15.11 | AO2 src max | 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*).<br><br> | |
| | -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 AO2 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 analog output AO3. | |
| | Speed rpm | *01.01 Motor speed rpm* (see page *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | Frequency | *01.03 Output frequency* (see page *112*). | 1073742083 |
| | Current | *01.04 Motor current* (see page *112*). | 1073742084 |
| | Current % | *01.05 Motor current %* (see page *112*). | 1073742085 |
| | Torque | *01.06 Motor torque* (see page *112*). | 1073742086 |
| | Dc-voltage | *01.07 Dc-voltage* (see page *112*). | 1073742087 |
| | Power inu | *01.22 Power inu out* (see page *112*). | 1073742102 |
| | Power motor | *01.23 Motor power* (see page *112*). | 1073742103 |
| | SpRef unramp | *03.03 SpeedRef unramp* (see page *123*). | 1073742595 |
| | SpRef ramped | *03.05 SpeedRef ramped* (see page *123*). | 1073742597 |
| | SpRef used | *03.06 SpeedRef used* (see page *123*). | 1073742598 |
| | TorqRef used | *03.14 Torq ref used* (see page *123*). | 1073742606 |
| | Proc PID out | *04.05 Process PID out* (see page *123*). | 1073742853 |
| | Process act% | *04.22 Act val %* (see page *124*). | 1073742870 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 15.14 | AO3 filt time | Defines the filtering time constant for analog 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 analog 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 analog output AO3. | |
| | 0.000 … 22.700 mA | Minimum AO3 output value. | 1000 = 1 mA |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 15.17 | AO3 src max | 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*).     | |
| | -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 |
| 15.25 | AO ctrl word | Defines how a signed source is processed before output. | |

| 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 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| **16 System** | | Local lock and parameter lock settings; parameter restore; user parameter set load/save; parameter change log reset; parameter list settings; unit of power selection; application macro display. | |
| 16.01 | Local lock | Selects the source for disabling local control (Take/Release button in the PC tool, LOC/REM key of the panel).<br>0 = Local control enabled.<br>1 = Local control disabled.<br>**WARNING!** Before activating, ensure that the control panel is not needed for stopping the drive! | |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | 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 | Selects the pass code for the parameter lock (see parameter *16.02 Parameter lock*).<br>After entering 358 at this parameter, parameter *16.02 Parameter lock* can be adjusted. The value reverts back to 0 automatically. | |
| | 0 … 2147483647 | Pass code for parameter lock. | 1 = 1 |
| 16.04 | Param restore | Restores the original settings of the application, i.e. parameter factory default values.<br>**Note:** This parameter cannot be changed while the drive is running. | |
| | Done | Restoring is completed. | 0 |
| | Restore defs | All parameter values are restored to default values, except motor data, ID run results, and fieldbus adapter and drive-to-drive link configuration data. | 1 |
| | Clear all | All parameter values are restored to default values, including motor data, ID run results, and fieldbus adapter and drive-to-drive link configuration data. PC tool communication is interrupted during the restoring. Drive CPU is re-booted after the restoring is completed. | 2 |
| 16.07 | Param save | Saves the valid parameter values to the permanent memory.<br>**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 |

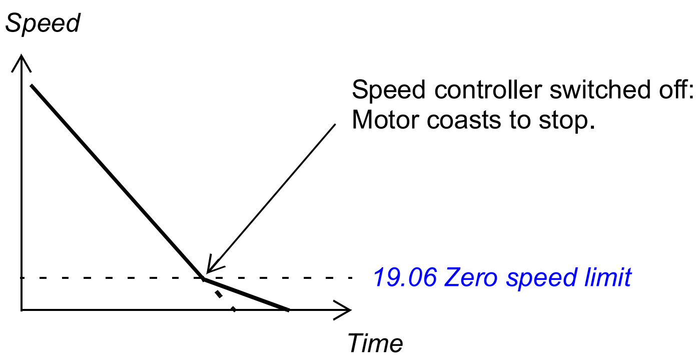
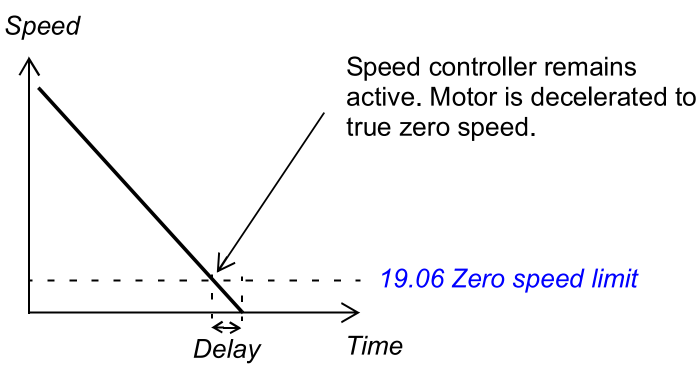
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 16.09 | User set sel | Enables the saving and restoring of up to four custom sets of parameter settings.<br>The set that was in use before powering down the drive is in use after the next power-up.<br>**Notes:**<br>• Fieldbus adapter parameters (groups 50…53) are not part of user parameter sets.<br>• 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 |
| | Set4 par act | User parameter set 4 has been loaded using parameter *16.09 User set sel*. | 1024 |

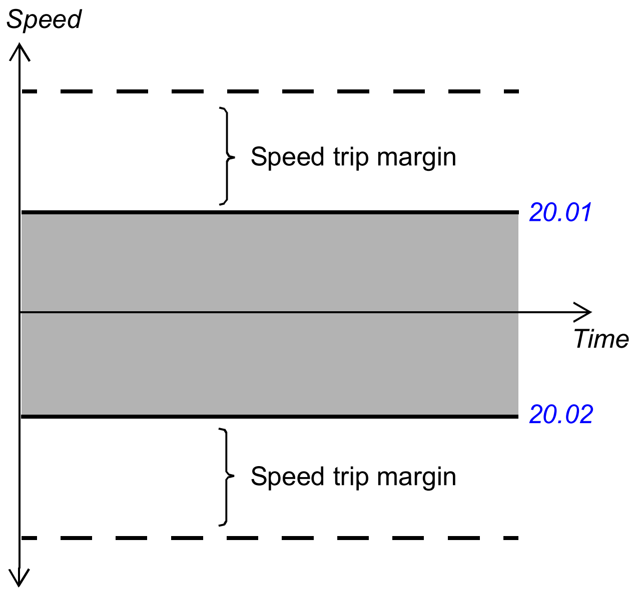
<table>
<tr><th>No.</th><th>Name/Value</th><th>Description</th><th>FbEq</th></tr>
<tr><td>16.11</td><td>User IO sel lo</td><td>When parameter 16.09 User set sel is set to IO mode, selects the user parameter set together with parameter 16.12 User IO sel hi. The status of the source defined by this parameter and parameter 16.12 select the user parameter set as follows:
<table>
<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>
<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>
</table></td><td></td></tr>
<tr><td></td><td>Const</td><td rowspan="2">Bit pointer setting (see Terms and abbreviations on page 109).</td><td rowspan="2">-</td></tr>
<tr><td></td><td>Pointer</td></tr>
<tr><td>16.12</td><td>User IO sel hi</td><td>See parameter 16.11 User IO sel lo.</td><td></td></tr>
<tr><td></td><td>Const</td><td rowspan="2">Bit pointer setting (see Terms and abbreviations on page 109).</td><td rowspan="2">-</td></tr>
<tr><td></td><td>Pointer</td></tr>
<tr><td>16.14</td><td>Reset ChgParLog</td><td>Resets the log of latest parameter changes.</td><td></td></tr>
<tr><td></td><td>Done</td><td>Reset not requested (normal operation).</td><td>0</td></tr>
<tr><td></td><td>Reset</td><td>Reset log of latest parameter changes. The value reverts automatically to Done.</td><td>1</td></tr>
<tr><td>16.16</td><td>Menu set active</td><td>Shows which parameter list is active. Parameter lists determine which parameters are displayed.<br>See also parameter 16.21 Menu selection.</td><td></td></tr>
<tr><td></td><td>None</td><td>No specific parameter list is active.</td><td>0</td></tr>
<tr><td></td><td>Single short</td><td>A selective list of parameters relevant to the single pump (factory default) application macro is displayed.</td><td>1</td></tr>
<tr><td></td><td>Single long</td><td>A more comprehensive list of parameters relevant to the single pump (factory default) application macro is displayed.</td><td>2</td></tr>
<tr><td></td><td>Trad short</td><td>A selective list of parameters relevant to the traditional pump control application macro is displayed.</td><td>3</td></tr>
<tr><td></td><td>Trad long</td><td>A more comprehensive list of parameters relevant to the traditional pump control application macro is displayed.</td><td>4</td></tr>
<tr><td></td><td>Ext short</td><td>A selective list of parameters relevant to the external control application macro is displayed.</td><td>5</td></tr>
<tr><td></td><td>Ext long</td><td>A more comprehensive list of parameters relevant to the external control application macro is displayed.</td><td>6</td></tr>
<tr><td></td><td>H/A short</td><td>A selective list of parameters relevant to the Hand/Auto control application macro is displayed.</td><td>7</td></tr>
<tr><td></td><td>H/A long</td><td>A more comprehensive list of parameters relevant to the Hand/Auto control application macro is displayed.</td><td>8</td></tr>
<tr><td></td><td>Level short</td><td>A selective list of parameters relevant to the Level control application macro (single-pump) is displayed.</td><td>9</td></tr>
<tr><td></td><td>Level long</td><td>A more comprehensive list of parameters relevant to the Level control application macro (single-pump) is displayed.</td><td>10</td></tr>
</table>

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | M lvl short | A selective list of parameters relevant to the Level control application macro (multipump) is displayed. | 11 |
| | M lvl long | A more comprehensive list of parameters relevant to the Level control application macro (multipump) is displayed. | 12 |
| | M pump short | A selective list of parameters relevant to the Multipump control application macro (single-pump) is displayed. | 13 |
| | M pump long | A more comprehensive list of parameters relevant to the Multipump control application macro (single-pump) is displayed. | 14 |
| | Full | All parameters are displayed. | 15 |
| 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 |
| 16.20 | Macro selected | Shows which application macro is currently selected. For more information, see chapter *Application macros* (page *87*).<br>**Note:** Changing the value of this parameter does not change the current application macro. To change the application macro, use the Application macro assistant available through the control panel instead. | |
| | Factory def | Factory default macro. | 0 |
| | Ext ctrl | External control macro. | 1 |
| | Trad ctrl | Traditional pump control macro. | 2 |
| | Hand/Auto | Hand/Auto macro. | 3 |
| | Level ctrl | Level control macro (for a single pump). | 4 |
| | Multi level | Level control macro (for multiple pumps). | 5 |
| | Multi pump | Multipump control macro. | 6 |
| 16.21 | Menu selection | Loads a short, long or full parameter list. | |
| | Short | Only a selective list of parameters will be displayed. | 0 |
| | Long | Only the parameters relevant to the current application macro are displayed. | 1 |
| | Full | All parameters are displayed, including those not relevant to the current application macro. | 2 |

| | | | |
|---|---|---|---|
| **19 Speed calculation** | | Speed scaling, feedback and supervision 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 |
| 19.02 | Speed fb sel | Selects the speed feedback value used in control.<br>**Note:** The speed feedback value is always estimated. | |
| | Estimated | A calculated speed estimate is used. | 0 |

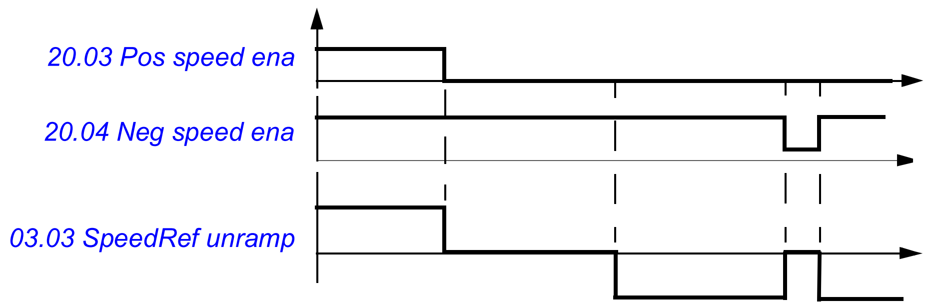
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 19.03 | MotorSpeed filt | Defines the time constant of the actual speed filter, i.e. time within the actual speed has reached 63% of the nominal speed (filtered speed = *01.01 Motor speed rpm*).<br>If the used speed reference remains constant, 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.<br>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<br>$t_{mech} = (n_{nom} / T_{nom}) \times J_{tot} \times 2\pi / 60$, where<br>$J_{tot}$ = total inertia of the load and motor (the gear ratio between the load and motor must be taken into account)<br>$n_{nom}$ = motor nominal speed<br>$T_{nom}$ = motor nominal torque<br>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 | 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.<br>**Without Zero speed delay:**<br>The drive receives a stop command and decelerates along a ramp. When the motor actual speed falls below the value of *19.06 Zero speed limit*, the speed controller is switched off. The inverter modulation is stopped and the motor coasts to standstill.<br><br><br>**With Zero speed delay:**<br>The drive receives a stop command and decelerates along a ramp. When the actual motor speed falls below the value of *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.<br><br> | |
| | 0 … 30000 ms | Zero speed delay. | 1 = 1 ms |
| 19.08 | Above speed lim | Defines the supervision limit for the actual speed. | |
| | 0 … 30000 rpm | Actual speed supervision limit. | 1 = 1 rpm |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 19.09 | Speed TripMargin | 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 (0x7310)* fault.<br>**Example:** If the maximum speed is 1420 rpm and speed trip margin is 300 rpm, the drive trips at 1720 rpm.<br> | |
| | 0.0 … 10000.0 rpm | Overspeed trip margin. | 10 = 1 rpm |
| 19.10 | Speed window | 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 unramp*). 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. | |
| | 0 … 30000 rpm | Absolute value for motor speed window supervision. | 1 = 1 rpm |

| | | | |
|---|---|---|---|
| **20 Limits** | | Drive operation limits.<br>See also section *Speed controller tuning* on page *68*. | |
| 20.01 | Maximum speed | Defines the allowed maximum speed. | |
| | 0 … 30000 rpm | Maximum speed. | 1 = 1 rpm |
| 20.02 | Minimum speed | Defines the allowed minimum speed.<br>**Note:** If the motor may only be run in the forward direction within a certain range above 0 rpm, leave this parameter at 0 rpm, and use parameter *21.09 SpeedRef min abs* to define the lower boundary of the range. | |
| | -30000 … 0 rpm | Minimum speed. | 1 = 1 rpm |

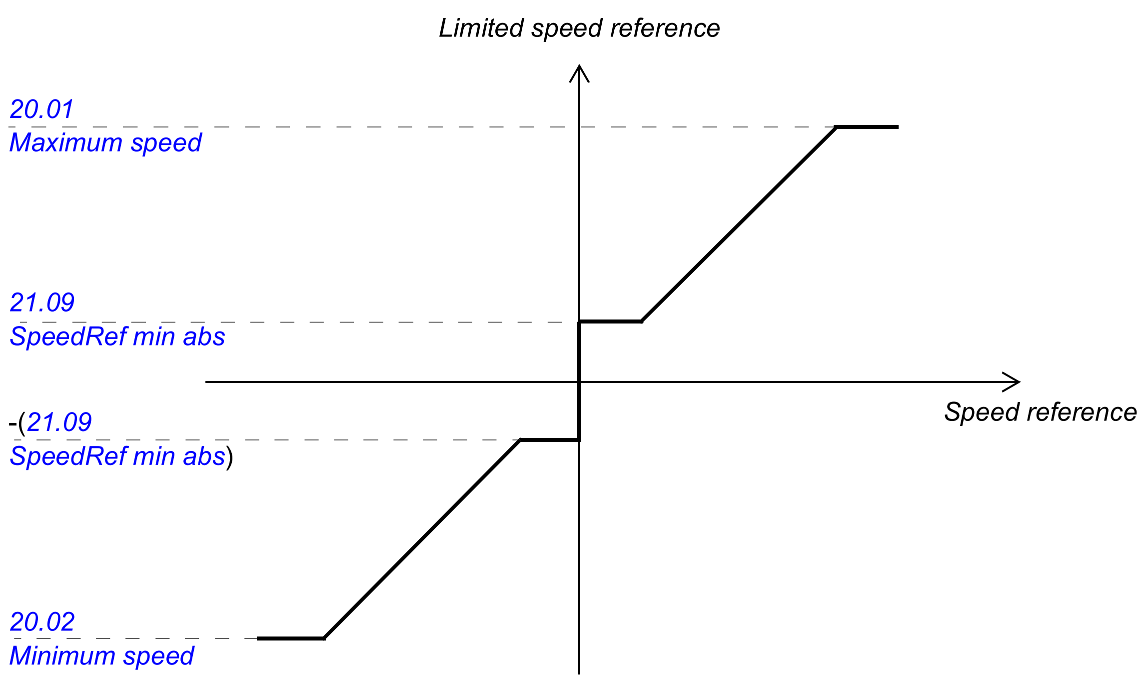
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 20.03 | Pos speed ena | Selects the source of the positive speed reference enable command.<br>1 = Positive speed reference is enabled.<br>0 = Positive speed reference is interpreted as zero speed reference (In the figure below *03.03 SpeedRef unramp* is set to zero after the positive speed enable signal has cleared). The speed reference is set to zero and the motor is stopped along the currently active deceleration ramp. | |




| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | | **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 a 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. | |
| | Const<br>Pointer | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 20.04 | Neg speed ena | Selects the source of the negative speed reference enable command. See parameter *20.03 Pos speed ena*. | |
| | Const<br>Pointer | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 20.05 | Maximum current | Defines the maximum allowed motor current. | |
| | 0.00 … 30000.00 A | Maximum motor current. | 100 = 1 A |
| 20.06 | Torq lim sel | Defines a source that selects between the two sets of torque limits defined by parameters *20.07…20.10*.<br>0 = The torque limits defined by parameters *20.07 Maximum torque1* and *20.08 Minimum torque1* are in force.<br>1 = The torque limits defined by parameters *20.09 Maximum torque2* and *20.10 Minimum torque2* are in force. | |
| | Const<br>Pointer | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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% |

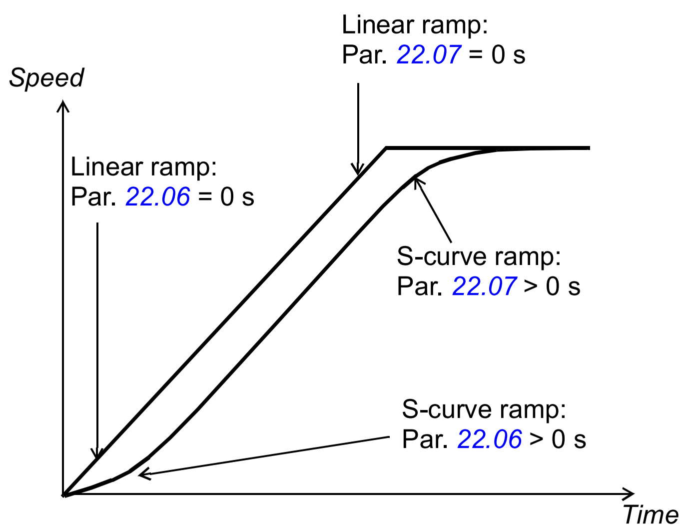
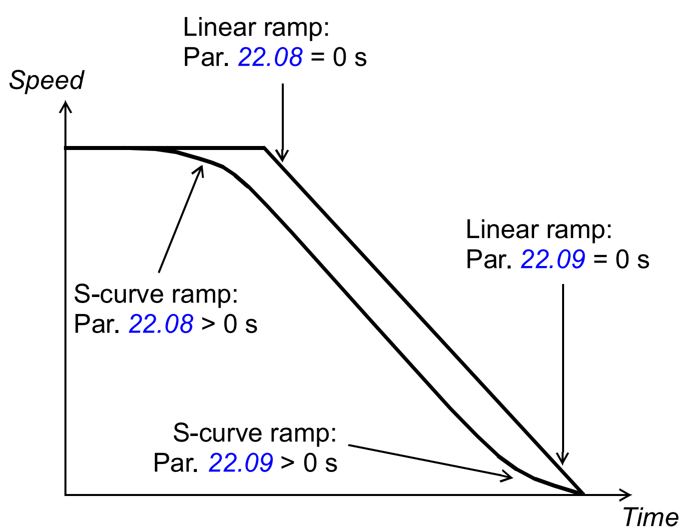
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 20.09 | Maximum torque2 | Defines the source of maximum torque limit 2 for the drive (in percent of the motor nominal torque). See parameter *20.06 Torq lim sel*. | |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | FBA ref1 | *02.26 FBA main ref1* (see page *118*). | 1073742362 |
| | FBA ref2 | *02.27 FBA main ref2* (see page *118*). | 1073742363 |
| | Max torque1 | *20.07 Maximum torque1* (see page *171*). | 1073746951 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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 *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | FBA ref1 | *02.26 FBA main ref1* (see page *118*). | 1073742362 |
| | FBA ref2 | *02.27 FBA main ref2* (see page *118*). | 1073742363 |
| | Neg max torq | -*20.09 Maximum torque2* (see page *172*). | 1073746949 |
| | Min torque1 | *20.08 Minimum torque1* (see page *171*). | 1073746952 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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. | |
| | 0.0 … 1600.0% | Maximum generating power. | 10 = 1% |

| **21 Speed ref** | | Speed reference source selection and processing. | |
|---|---|---|---|
| 21.01 | Speed ref1 sel | Selects the source for speed reference 1. | |
| | Zero | Zero speed reference. | 0 |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | Freq in | *02.20 Freq in* (see page *114*). | 1073742356 |
| | FBA ref1 | *02.26 FBA main ref1* (see page *118*). | 1073742362 |
| | FBA ref2 | *02.27 FBA main ref2* (see page *118*). | 1073742363 |
| | Panel | *02.34 Panel ref* (see page *118*). | 1073742370 |
| | EFB ref1 | *02.38 EFB main ref1* (see page *122*). | 1073742374 |
| | EFB ref2 | *02.39 EFB main ref2* (see page *122*). | 1073742375 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 21.02 | Speed ref2 sel | Selects the source for speed reference 2.<br>**Note:** The reference signal must be in the range 0…100. | |
| | Zero | Zero speed reference. | 0 |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |

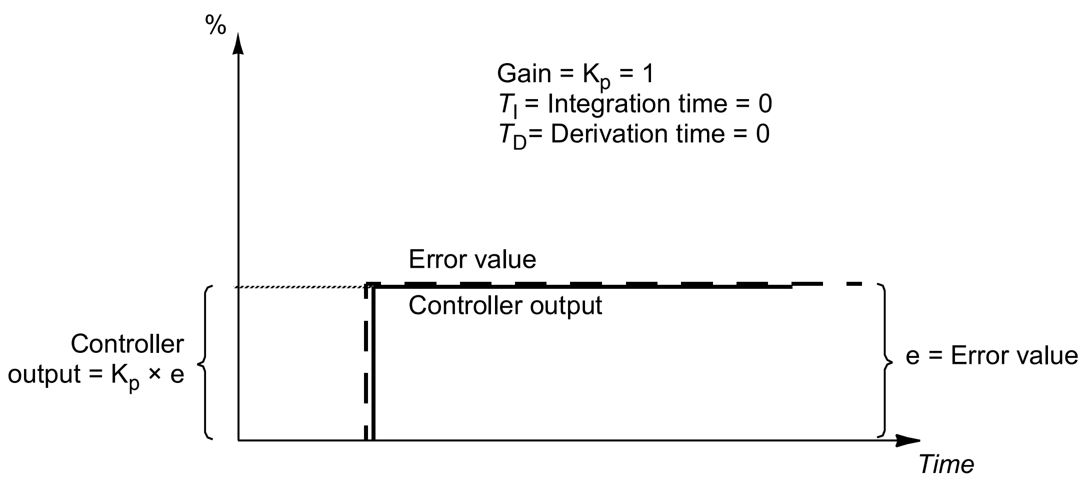
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | Freq in | *02.20 Freq in* (see page *114*). | 1073742356 |
| | FBA ref1 | *02.26 FBA main ref1* (see page *118*). | 1073742362 |
| | FBA ref2 | *02.27 FBA main ref2* (see page *118*). | 1073742363 |
| | Panel | *02.34 Panel ref* (see page *118*). | 1073742370 |
| | EFB ref1 | *02.38 EFB main ref1* (see page *122*). | 1073742374 |
| | EFB ref2 | *02.39 EFB main ref2* (see page *122*). | 1073742375 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 21.05 | Speed share | Defines the scaling factor for the speed reference (the speed reference is multiplied by the defined value). | |
| | -8.000 …8.000 | Speed reference scaling factor. | 1000 = 1 |
| 21.09 | SpeedRef min abs | Defines the absolute minimum limit for the speed reference.   | |
| | 0 … 30000 rpm | Absolute minimum limit for speed reference. | 1 = 1 rpm |

| **22 Speed ref ramp** | | Speed reference and emergency stop (OFF3) ramp settings. | |
|---|---|---|---|
| 22.02 | Acc time | Defines acceleration time as the time required for the speed to change from zero to the speed value defined by parameter *19.01 Speed scaling*.<br>If the speed reference increases faster than the set acceleration rate, the motor speed will follow the acceleration rate.<br>If the speed reference increases slower than the set acceleration rate, the motor speed will follow the reference signal.<br>If the acceleration time is set too short, the drive will automatically prolong the acceleration in order not to exceed the drive torque limits. | |
| | 0.000 … 1800.000 s | Acceleration time. | 1000 = 1 s |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 22.03 | Dec time | Defines deceleration time as the time required for the speed to change from the speed value defined by parameter *19.01 Speed scaling* to zero.<br>If the speed reference decreases slower than the set deceleration rate, the motor speed will follow the reference signal.<br>If the reference changes faster than the set deceleration rate, the motor speed will follow the deceleration rate.<br>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*). | |
| | 0.000 … 1800.000 s | Deceleration time. | 1000 = 1 s |

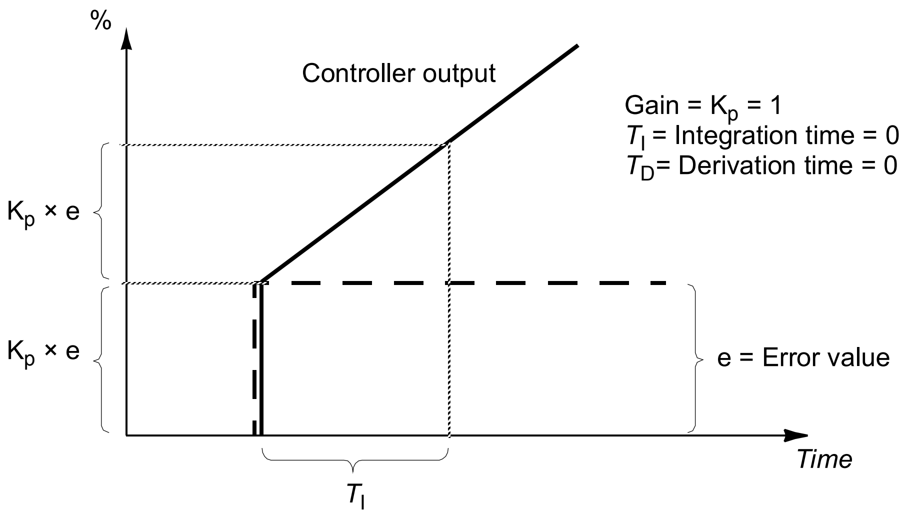
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 22.06 | Shape time acc1 | Defines the shape of the acceleration ramp at the beginning of the acceleration.<br>0.000 s: Linear ramp. Suitable for steady acceleration or deceleration and for slow ramps.<br>0.001…1000.000 s: S-curve ramp. S-curve 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.<br>**Acceleration:**<br><br>**Deceleration:**<br> | |
| | 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 |
| 22.12 | 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. | |
| 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. | |

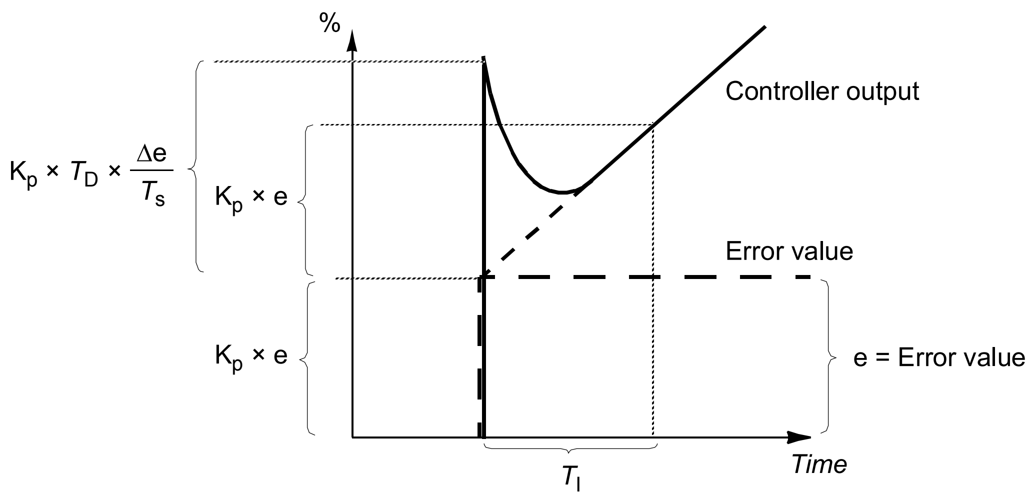


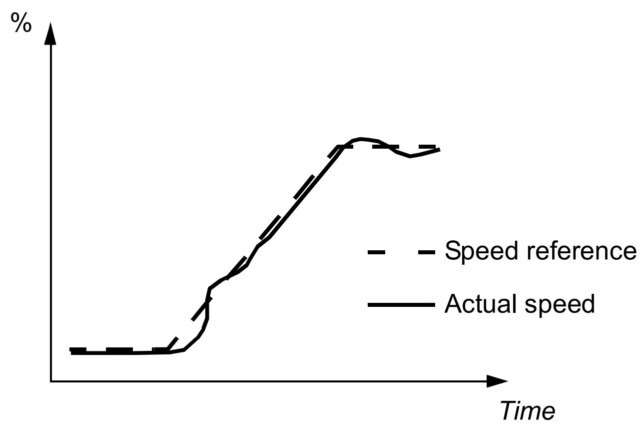
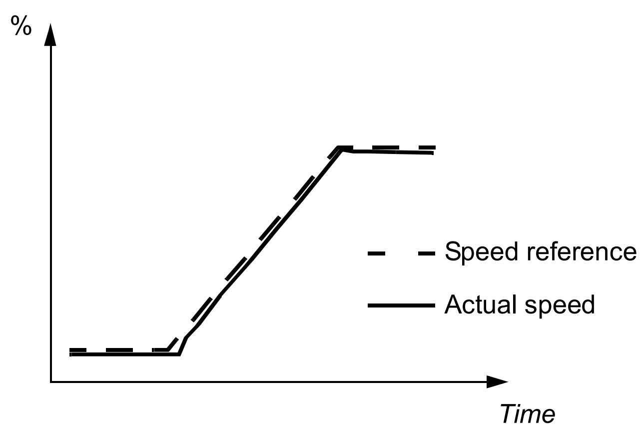

| | | | |
|---|---|---|---|
| | | If gain is set to 1, a 10% change in error value (reference - actual value) causes the speed controller output to change by 10%. | |
| | 0.00 … 200.00 | Proportional gain for speed controller. | 100 = 1 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 23.02 | Integration time | 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.<br>If parameter value is set to zero, the I-part of the controller is disabled.<br>Anti-windup stops the integrator if the controller output is limited. See *06.05 Limit word1*.<br>The figure below shows the speed controller output after an error step when the error remains constant. | |

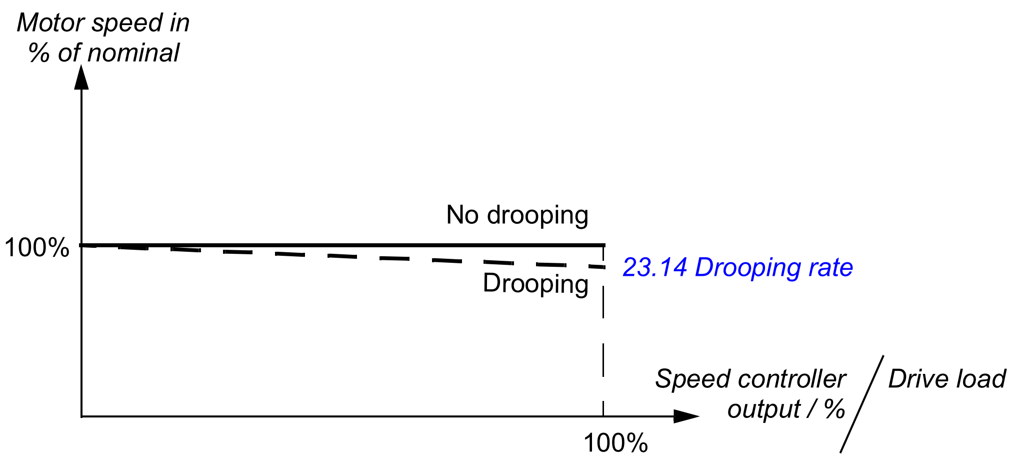



| | | | |
|---|---|---|---|
| | 0.000 … 600.000 s | Integration time for speed controller. | 1000 = 1 s |

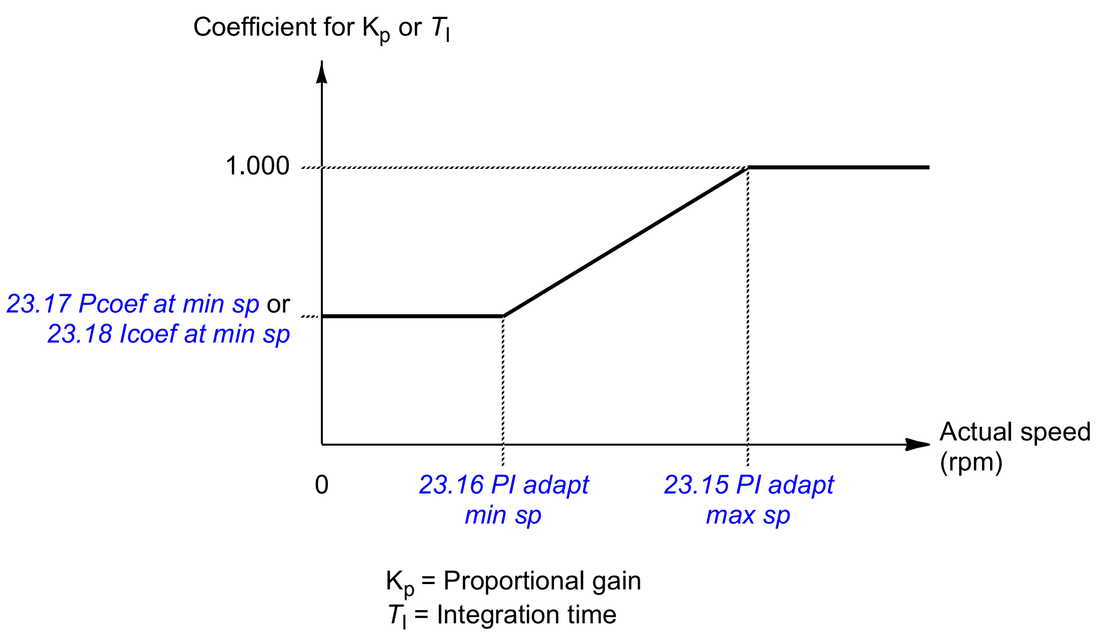
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 23.03 | Derivation time | Defines the derivation time of the speed controller. Derivative action boosts the controller output if the error value changes. The longer the derivation time, the more the speed controller output is boosted during 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.<br>The speed error derivative must be filtered with a low pass filter to eliminate disturbances.<br>The figure below shows the speed controller output after an error step when the error remains constant.<br><br>Gain = $K_p$ = 1<br>$T_I$ = Integration time > 0<br>$T_D$= Derivation time > 0<br>$T_s$= Sample time period = 250 µs<br>$\Delta$e = Error value change between two samples | |
| | 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 | 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*.<br>**Note:** As a general rule, set this parameter to a value between 50 and 100% of the sum of the mechanical time constants of the motor and the driven machine.<br>The figure below shows the speed responses when a high inertia load is accelerated along a ramp.<br>**No acceleration compensation:**<br><br>**Acceleration compensation:**<br> | |
| | 0.00 … 600.00 s | Acceleration compensation derivation time. | 100 = 1 s |
| 23.06 | Acc comp Ftime | Defines the derivation filter time constant for the acceleration(/deceleration) compensation. See parameters *23.03 Derivation time* and *23.05 Acc comp DerTime*. | |
| | 0.0 … 1000.0 ms | Derivation filter time constant for acceleration compensation. | 10 = 1 ms |
| 23.07 | Speed err Ftime | Defines the time constant of the speed error low pass filter.<br>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. | |
| | 0.0 … 1000.0 ms | Speed error filtering time constant. 0 = filtering disabled. | 10 = 1 ms |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 23.08 | Speed additive | Defines a speed reference to be added after ramping.<br>**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 *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | FBA ref1 | *02.26 FBA main ref1* (see page *118*). | 1073742362 |
| | FBA ref2 | *02.27 FBA main ref2* (see page *118*). | 1073742363 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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.<br>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<br>• the speed error exceeds the upper boundary of the window (parameter *23.12 SpeedErr win hi*), or<br>• the absolute value of the negative speed error exceeds the lower boundary of the window (*23.13 SpeedErr win lo*).<br>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.<br>**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 | Defines the droop rate in percent of the motor nominal speed. Drooping slightly decreases the drive speed as the drive load increases. The actual speed decrease at a certain operating point depends on the droop 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.<br>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.<br>The correct droop rate for a process must be found out case by case in practice.<br><br>**Speed decrease =** Speed controller output × Drooping × Max. speed<br>**Example:** Speed controller output is 50%, droop rate is 1%, maximum speed of the drive is 1500 rpm.<br>Speed decrease = 0.50 × 0.01 × 1500 rpm = 7.5 rpm.<br> | |
| | 0.00 … 100.00% | Droop rate. | 100 = 1% |

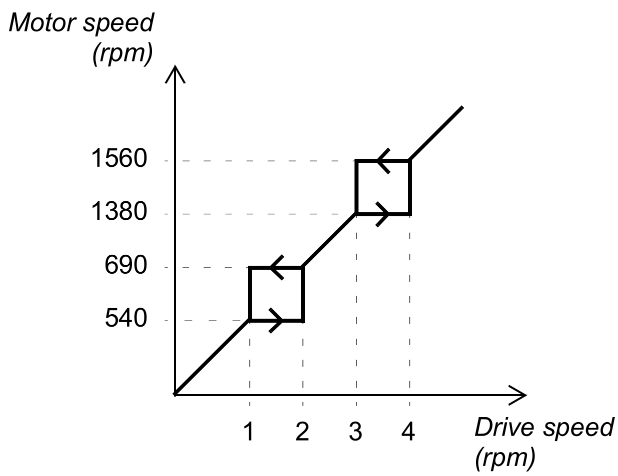
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 23.15 | PI adapt max sp | Maximum actual speed for speed controller adaptation.<br>Speed controller gain and integration time can be adapted according to actual speed. This is done by multiplying the gain (*23.01 Proport gain*) and integration time (*23.02 Integration time*) by coefficients at certain speeds. The coefficients are defined individually for both gain and integration time.<br>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 Icoef at min sp* respectively.<br>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.<br>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. | |




| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 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 | Icoef 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 | Activates the speed controller autotune function.<br>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 *User* autotune mode is chosen, also *23.07 Speed err Ftime* is automatically set.<br>The status of the autotune routine is shown by parameter *06.03 Speed ctrl stat*.<br>**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!<br>**Notes:**<br>• Before using the autotune function, the following parameters should be set:<br>  • All parameters adjusted during the start-up as described in the *ACQ810-04 drive modules Start-up Guide*<br>  • *19.01 Speed scaling*<br>  • *19.03 MotorSpeed filt*<br>  • *19.06 Zero speed limit*<br>  • Speed reference ramp settings in group *22 Speed ref ramp*<br>  • *23.07 Speed err Ftime*.<br>• The drive must be in local control mode and stopped before an autotune is requested.<br>• After requesting an autotune with this parameter, start the drive within 20 seconds.<br>• Wait until the autotune routine is completed (this parameter has reverted to the value *Done*). The routine can be aborted by stopping the drive.<br>• Check the values of the parameters set by the autotune function.<br>See also section *Speed controller tuning* on page *68*. | |
| | Done | No tuning has been requested (normal operation) | 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 after autotune procedure in user mode. A larger bandwidth results in more restricted speed controller settings. | |
| | 0.00 … 2000.00 Hz | Tune bandwidth for user PI tune mode. | 100 = 1 Hz |
| 23.22 | Tune damping | Speed controller damping after autotune procedure in user mode. Higher damping results in safer and smoother operation. | |
| | 0.0 … 200.0 | Speed controller damping for user PI tune mode. | 10 = 1 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| **25 Critical speed** | | Configuration of critical speeds (or ranges of speed) that are avoided due to, for example, mechanical resonance problems. | |
| 25.01 | Crit speed sel | Enables/disables the critical speeds function.<br>**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:<br>• activate the critical speeds function,<br>• set the critical speed ranges as in the figure below. | |




| | |
|---|---|
| 1 | Par. *25.02* = 540 rpm |
| 2 | Par. *25.03* = 690 rpm |
| 3 | Par. *25.04* = 1380 rpm |
| 4 | Par. *25.05* = 1590 rpm |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 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.<br>**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.<br>**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 |
| 25.04 | Crit speed2 lo | Defines the low limit for critical speed range 2.<br>**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 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 25.05 | Crit speed2 hi | Defines the high limit for critical speed range 2.<br>**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.<br>**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.<br>**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.<br>An active constant speed overrides the drive speed reference. See also section *Constant speeds* on page *68*. | |
|---|---|---|---|
| 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*). |

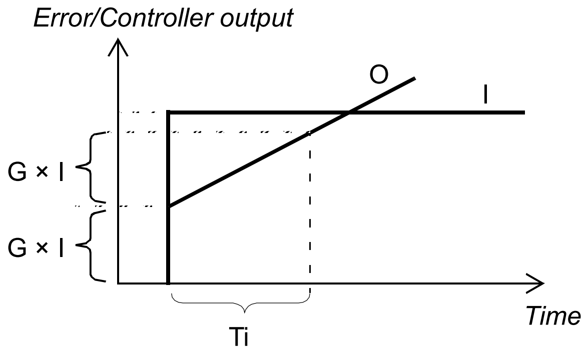
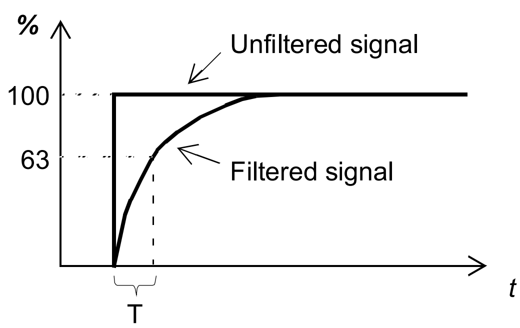
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 26.02 | Const speed sel1 | When bit 0 of parameter *26.01 Const speed func* is 0 (Separate), selects a source that activates constant speed 1.<br>When bit 0 of parameter *26.01 Const speed func* is 1 (Packed), this parameter and parameters *26.03 Const speed sel2* and *26.04 Const speed sel3* select three sources whose states activate constant speeds as follows: | |

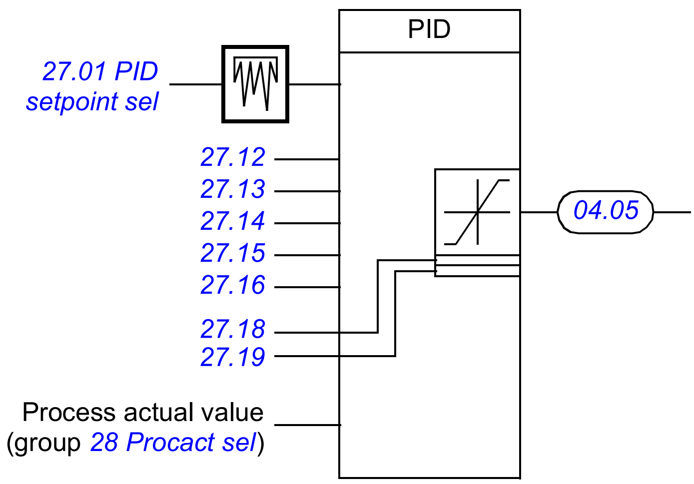
| Source defined by par. ***26.02*** | Source defined by par. ***26.03*** | Source defined by par. ***26.04*** | Constant speed active |
|---|---|---|---|
| 0 | 0 | 0 | None |
| 1 | 0 | 0 | Constant speed 1 |
| 0 | 1 | 0 | Constant speed 2 |
| 1 | 1 | 0 | Constant speed 3 |
| 0 | 0 | 1 | Constant speed 4 |
| 1 | 0 | 1 | Constant speed 5 |
| 0 | 1 | 1 | Constant speed 6 |
| 1 | 1 | 1 | Constant speed 7 |

| 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 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.<br>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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 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.<br>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 |
| | DI2 | Digital input DI2 (as indicated by *02.01 DI status*, bit 1). | 1073807873 |

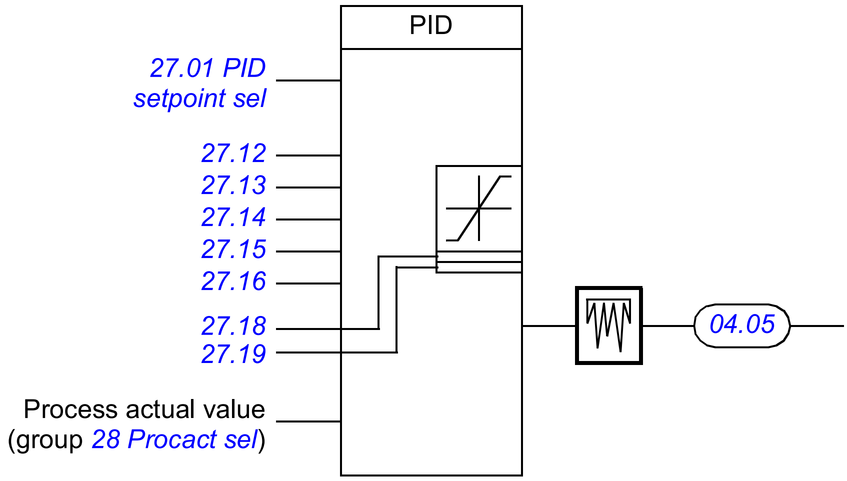
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | DI3 | Digital input DI3 (as indicated by *02.01 DI status*, bit 2). | 1073873409 |
| | DI4 | Digital input DI4 (as indicated by *02.01 DI status*, bit 3). | 1073938945 |
| | DI5 | Digital input DI5 (as indicated by *02.01 DI status*, bit 4). | 1074004481 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 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.<br>See also section *PID control* on page *58*. | |
| 27.01 | PID setpoint sel | Selects the source of setpoint (reference) for the PID controller. | |
| | Zero | Zero reference. | 0 |
| | Setpoint % | *04.25 Setpoint val %* (see page *124*). | 1073742873 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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 |

<table>
<tr><th>No.</th><th>Name/Value</th><th>Description</th><th>FbEq</th></tr>
<tr><td>27.13</td><td>PID integ time</td><td>Defines the integration time for the process PID controller.<br><br><br>I = controller input (error)<br>O = controller output<br>G = gain<br>Ti = integration time</td><td></td></tr>
<tr><td></td><td>0.00 … 320.00 s</td><td>Integration time.</td><td>100 = 1 s</td></tr>
<tr><td>27.14</td><td>PID deriv time</td><td>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:<br>PID DERIV TIME × ($E_K$ - $E_{K-1}$)/$T_S$, in which<br>$T_S$ = 12 ms sample time<br>E = Error = Process setpoint - process actual value.</td><td></td></tr>
<tr><td></td><td>0.00 … 10.00 s</td><td>Derivation time.</td><td>100 = 1 s</td></tr>
<tr><td>27.15</td><td>PID deriv filter</td><td>Defines the time constant of the 1-pole filter used to smooth the derivative component of the process PID controller.<br><br><br>$O = I \times (1 - e^{-t/T})$<br>I = filter input (step)<br>O = filter output<br>t = time<br>T = filter time constant</td><td></td></tr>
<tr><td></td><td>0.00 … 10.00 s</td><td>Filter time constant.</td><td>100 = 1 s</td></tr>
</table>

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 27.16 | PID error inv | PID error inversion. When the source selected by this parameter is on, the error (process setpoint – process actual value) at the PID controller input is inverted. | |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 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 |
| 27.30 | Pid ref freeze | Freezes, or defines a source that can be used to freeze, the setpoint (reference) input of the process PID controller. This feature is useful when the reference is based on a process feedback connected to an analog input, and the sensor must be serviced without stopping the process.<br>The setpoint input of the PID controller is frozen as long as the selected source is 1.<br>See also parameter *27.31 Pid out freeze*.<br> | |
| | No | Process PID controller input not frozen. | 0 |
| | Freeze | Process PID controller input frozen. | 1 |
| | DI1 | Activation of digital input DI1 (as indicated by *02.01 DI status*, bit 0) freezes process PID controller input. | 1073742337 |
| | DI2 | Activation of digital input DI2 (as indicated by *02.01 DI status*, bit 1) freezes process PID controller input. | 1073807873 |
| | DI3 | Activation of digital input DI3 (as indicated by *02.01 DI status*, bit 2) freezes process PID controller input. | 1073873409 |
| | DI4 | Activation of digital input DI4 (as indicated by *02.01 DI status*, bit 3) freezes process PID controller input. | 1073938945 |
| | DI5 | Activation of digital input DI5 (as indicated by *02.01 DI status*, bit 4) freezes process PID controller input. | 1074004481 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 27.31 | Pid out freeze | Freezes, or defines a source that can be used to freeze, the output of the process PID controller. This feature can be used when, for example, a sensor providing process feedback must be serviced without stopping the process.<br>The output of the PID controller is frozen as long as the selected source is 1.<br>See also parameter *27.30 Pid ref freeze*. | |




| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | No | Process PID controller output not frozen. | 0 |
| | Freeze | Process PID controller output frozen. | 1 |
| | DI1 | Activation of digital input DI1 (as indicated by *02.01 DI status*, bit 0) freezes process PID controller output. | 1073742337 |
| | DI2 | Activation of digital input DI2 (as indicated by *02.01 DI status*, bit 1) freezes process PID controller output. | 1073807873 |
| | DI3 | Activation of digital input DI3 (as indicated by *02.01 DI status*, bit 2) freezes process PID controller output. | 1073873409 |
| | DI4 | Activation of digital input DI4 (as indicated by *02.01 DI status*, bit 3) freezes process PID controller output. | 1073938945 |
| | DI5 | Activation of digital input DI5 (as indicated by *02.01 DI status*, bit 4) freezes process PID controller output. | 1074004481 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 27.32 | Pipefill ref acc | Defines the time for the PID setpoint increase from 0 to 100%. | |
| | 0 … 100 s | PID setpoint acceleration time. | 1 = 1 s |
| 27.33 | Pipefill ref dec | Defines the time for the PID setpoint decrease from 100 to 0%. | |
| | 0 … 100 s | PID setpoint deceleration time. | 1 = 1 s |
| 27.34 | PID bal ena | Selects a source that enables the PID balancing reference (see parameter *27.35 PID bal ref*).<br>1 = PID balancing reference 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 |
| | 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 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | DI5 | Digital input DI5 (as indicated by *02.01 DI status*, bit 4). | 1074004481 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 27.35 | PID bal ref | Defines the PID balancing reference. The PID controller output is set to this value when the source selected by parameter *27.35 PID bal ref* is 1. | |
| | -32768.0 … 32768.0% | PID balancing reference. | 10 = 1% |
| 27.36 | Pump scal speed | Defines pump speed that corresponds to 100% PID controller output. | |
| | Speed scal | *19.01 Speed scaling* (see page *167*). | 1073746689 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |

| | | | |
|---|---|---|---|
| **28 Procact sel** | | Process actual value (feedback) settings. | |
| 28.01 | Act val 1/2 sel | Selects the process actual value (1 or 2). Alternatively, selects a source whose status determines which process actual value is used (0 = Actual value 1; 1 = Actual value 2).<br>**Note:** This parameter is only effective when parameter *28.04 Act val func* is set to *Act1*. | |
| | Act val 1 | Process actual value 1 selected. | 0 |
| | Act val 2 | Process actual value 2 selected. | 1 |
| | DI1 | Status of digital input DI1 (as indicated by *02.01 DI status*, bit 0) determines which process actual value is selected. | 1073742337 |
| | DI2 | Status of digital input DI2 (as indicated by *02.01 DI status*, bit 1) determines which process actual value is selected. | 1073807873 |
| | DI3 | Status of digital input DI3 (as indicated by *02.01 DI status*, bit 2) determines which process actual value is selected. | 1073873409 |
| | DI4 | Status of digital input DI4 (as indicated by *02.01 DI status*, bit 3) determines which process actual value is selected. | 1073938945 |
| | DI5 | Status of digital input DI5 (as indicated by *02.01 DI status*, bit 4) determines which process actual value is selected. | 1074004481 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 28.02 | Act val 1 src | Selects the source of process actual value 1. | |
| | Zero | No source selected. | 0 |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | AI3 scaled | *02.09 AI3 scaled* (see page *113*). | 1073742345 |
| | AI4 scaled | *02.11 AI4 scaled* (see page *114*). | 1073742347 |
| | AI5 scaled | *02.13 AI5 scaled* (see page *114*). | 1073742349 |
| | FBA procact | *02.41 FBA act val* (see page *122*). | 1073742377 |
| | Shared sig1 | *02.43 Shared signal 1* (see page *122*). | 1073742379 |
| | Flow act | *05.05 Flow act* (see page *125*). | 1073743109 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 28.03 | Act val 2 src | Selects the source of process actual value 2. | |
| | Zero | No source selected. | 0 |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | AI3 scaled | *02.09 AI3 scaled* (see page *113*). | 1073742345 |
| | AI4 scaled | *02.11 AI4 scaled* (see page *114*). | 1073742347 |
| | AI5 scaled | *02.13 AI5 scaled* (see page *114*). | 1073742349 |
| | FBA procact | *02.41 FBA act val* (see page *122*). | 1073742377 |
| | Shared sig1 | *02.43 Shared signal 1* (see page *122*). | 1073742379 |
| | Flow act | *05.05 Flow act* (see page *125*). | 1073743109 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 28.04 | Act val func | Defines how the final process actual value is calculated from the two sources selected by parameters *28.02 Act val 1 src* and *28.03 Act val 2 src*. | |
| | Act1 | The actual value is determined by parameter *28.01 Act val 1/2 sel*. | 0 |
| | Add | Sum of actual value 1 and actual value 2. | 1 |
| | Sub | Actual value 2 subtracted from actual value 1. | 2 |
| | Mul | Actual value 1 multiplied by actual value 2. | 3 |
| | Div | Actual value 1 divided by actual value 2. | 4 |
| | Max | Greater of the two actual values used. | 5 |
| | Min | Smaller of the two actual values used. | 6 |
| | Sqrt sub | Square root of (actual value 1 – actual value 2). | 7 |
| | Sqrt add | Square root of actual value 1 + square root of actual value 2. | 8 |
| 28.05 | Act max val | Actual value scaling. The setting equals 100% of process setpoint and is typically set to the value that corresponds to the top end of the sensor range. | |
| | 0.00 … 32768.00% | Actual value scaling. | 100 = 1% |
| 28.06 | Act unit sel | Defines the unit for both process actual value and process setpoint. Typically the measured quantity is selected. | |
| | % | % | 4 |
| | m3/h | $m^3/h$ | 20 |
| | bar | bar | 22 |
| | kPa | kPa | 23 |
| | GPM | GPM | 24 |
| | psi | psi | 25 |
| | inHg | inHg | 29 |
| | mbar | mbar | 44 |
| | Pa | Pa | 45 |
| | inH2O | $inH_2O$ | 58 |
| | in wg | in wg | 59 |
| | ft wg | ft wg | 60 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | lbsi | lbsi | 61 |
| | m | m | 72 |
| | inch | inch | 73 |
| 28.07 | Act FBA scaling | Defines a divisor for process actual value for fieldbus. This parameter can be used to improve calculation accuracy at low and high values. | |
| | Not used | No scaling applied. | 0 |
| | Src/10 | The actual value is divided by 10 for fieldbus. | 1 |
| | Src/100 | The actual value is divided by 100 for fieldbus. | 2 |
| | Src/1000 | The actual value is divided by 1000 for fieldbus. | 3 |
| **29 Setpoint sel** | | Process setpoint (reference) settings. | |
| 29.01 | Setpoint 1 / 2 sel | Selects the process setpoint (1 or 2). Alternatively, selects a source whose status determines which process setpoint is used (0 = Setpoint 1; 1 = Setpoint 2). | |
| | Setpoint 1 | Setpoint 1 selected. | 0 |
| | Setpoint 2 | Setpoint 2 selected. | 1 |
| | DI1 | Status of digital input DI1 (as indicated by *02.01 DI status*, bit 0) determines which process setpoint is selected. | 1073742337 |
| | DI2 | Status of digital input DI2 (as indicated by *02.01 DI status*, bit 1) determines which process setpoint is selected. | 1073807873 |
| | DI3 | Status of digital input DI3 (as indicated by *02.01 DI status*, bit 2) determines which process setpoint is selected. | 1073873409 |
| | DI4 | Status of digital input DI4 (as indicated by *02.01 DI status*, bit 3) determines which process setpoint is selected. | 1073938945 |
| | DI5 | Status of digital input DI5 (as indicated by *02.01 DI status*, bit 4) determines which process setpoint is selected. | 1074004481 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 29.02 | Setpoint 1 src | Selects the source of process setpoint 1. | |
| | Zero | No source selected. | 0 |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | AI3 scaled | *02.09 AI3 scaled* (see page *113*). | 1073742345 |
| | AI4 scaled | *02.11 AI4 scaled* (see page *114*). | 1073742347 |
| | AI5 scaled | *02.13 AI5 scaled* (see page *114*). | 1073742349 |
| | FBA setpoint | *02.40 FBA setpoint* (see page *122*). | 1073742376 |
| | Shared sig2 | *02.44 Shared signal 2* (see page *122*). | 1073742380 |
| | Int set 1 | *29.04 Internal set 1* (see below). | 1073749252 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 29.03 | Setpoint 2 src | Selects the source of process setpoint 2. | |
| | Zero | No source selected. | 0 |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | AI3 scaled | *02.09 AI3 scaled* (see page *113*). | 1073742345 |
| | AI4 scaled | *02.11 AI4 scaled* (see page *114*). | 1073742347 |
| | AI5 scaled | *02.13 AI5 scaled* (see page *114*). | 1073742349 |
| | FBA setpoint | *02.40 FBA setpoint* (see page *122*). | 1073742376 |
| | Shared sig2 | *02.44 Shared signal 2* (see page *122*). | 1073742380 |
| | Int set 2 | *29.05 Internal set 2* (see below). | 1073749253 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 29.04 | Internal set 1 | Defines process setpoint 1 when parameter *29.02 Setpoint 1 src* is set to *Int set 1*. | |
| | 0.00 … 32768.00% | Internal process setpoint 1. | 100 = 1% |
| 29.05 | Internal set 2 | Defines process setpoint 2 when parameter *29.03 Setpoint 2 src* is set to *Int set 2*. | |
| | 0.00 … 32768.00% | Internal process setpoint 2. | 100 = 1% |
| 29.06 | Reference step 1 | Sets a percentage that is added to the process setpoint when one auxiliary (direct-on-line) motor is running.<br>*Example:* The drive operates three parallel pumps that pump water into a pipe. The pressure in the pipe is controlled. The constant pressure reference is set by parameter *29.04 Internal set 1*. During low water consumption, only the speed-regulated pump is run. When water consumption increases, constant-speed (direct-on-line) pumps are started: first one pump, and if the demand grows further, also the other pump. As water flow increases, the pressure loss between the beginning (point of measurement) and the end of the pipe increases. By setting suitable reference steps, the process setpoint is increased along with the increasing pumping capacity. The reference steps compensate the growing pressure loss and prevent the pressure fall at the end of the pipe. | |
| | 0.00 … 100.00% | Reference step 1. | 100 = 1% |
| 29.07 | Reference step 2 | Sets a percentage that is added to the process setpoint when two auxiliary (direct-on-line) motors are running. See parameter *29.06 Reference step 1*. | |
| | 0.00 … 100.00% | Reference step 2. | 100 = 1% |
| 29.08 | Reference step 3 | Sets a percentage that is added to the process setpoint when three auxiliary (direct-on-line) motors are running. See parameter *29.06 Reference step 1*. | |
| | 0.00 … 100.00% | Reference step 3. | 100 = 1% |
| 29.09 | Reference step 4 | Sets a percentage that is added to the process setpoint when four auxiliary (direct-on-line) motors are running. See parameter *29.06 Reference step 1*. | |
| | 0.00 … 100.00% | Reference step 4. | 100 = 1% |
| 29.10 | Reference step 5 | Sets a percentage that is added to the process setpoint when five auxiliary (direct-on-line) motors are running. See parameter *29.06 Reference step 1*. | |
| | 0.00 … 100.00% | Reference step 5. | 100 = 1% |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 29.11 | Reference step 6 | Sets a percentage that is added to the process setpoint when six auxiliary (direct-on-line) motors are running. See parameter *29.06 Reference step 1*. | |
| | 0.00 … 100.00% | Reference step 6. | 100 = 1% |
| 29.12 | Reference step 7 | Sets a percentage that is added to the process setpoint when seven auxiliary (direct-on-line) motors are running. See parameter *29.06 Reference step 1*. | |
| | 0.00 … 100.00% | Reference step 7. | 100 = 1% |
| **30 Fault functions** | | Configuration of behavior of the drive upon various fault situations. | |
| 30.01 | External fault | Selects a source for an external fault signal.<br>0 = External fault trip<br>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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | 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 (0x5300)*. | 1 |
| | Spd ref Safe | The drive generates alarm *LOCAL CTRL LOSS (0x5300)* and sets the speed to the speed defined by parameter *30.02 Speed ref safe*.<br>**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 (0x5300)* 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.<br>**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 (0x3182)*. | 1 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 (0x2330)*. | 1 |
| | Fault | The drive trips on fault *EARTH FAULT (0x2330)*. | 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 (0x3130)*. | 1 |
| 30.07 | Sto diagnostic | Selects how the drive reacts when it detects the absence of one or both Safe torque off (STO) signals.<br>**Note:** This parameter is for supervision only. The Safe torque off function can activate even when this parameter is set to *No*.<br>For general 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]). | |
| | Fault | The drive trips on *SAFE TORQUE OFF (0xFF7A)* if one or both of the STO signals are lost. | 1 |
| | Alarm | Drive running:<br>The drive trips on *SAFE TORQUE OFF (0xFF7A)* if one or both of the STO signals are lost.<br>Drive stopped:<br>The drive generates a *SAFE TORQUE OFF (0xFF7A)* alarm if both STO signals are absent. If only one of the signals is lost, the drive trips on *STO1 LOST (0x8182)* or *STO2 LOST (0x8183)*. | 2 |
| | No | Drive running:<br>The drive trips on *SAFE TORQUE OFF (0xFF7A)* if one or both of the STO signals are lost.<br>Drive stopped:<br>No action if both STO signals are absent. If only one of the signals is lost, the drive trips on *STO1 LOST (0x8182)* or *STO2 LOST (0x8183)*. | 3 |
| | Only Alarm | The drive generates a *SAFE TORQUE OFF (0xFF7A)* alarm if both STO signals are absent. If only one of the signals is lost, the drive trips on *STO1 LOST (0x8182)* or *STO2 LOST (0x8183)*. | 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 (0x3181)*. | 1 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 30.09 | Stall function | Selects how the drive reacts to a motor stall condition.<br>A stall condition is defined as follows:<br>• The drive is at stall current limit (*30.10 Stall curr lim*), and<br>• the output frequency is below the level set by parameter *30.11 Stall freq hi*, and<br>• the conditions above have been valid longer than the time set by parameter *30.12 Stall time*.<br>See section *Stall protection (parameters 30.09…30.12)* on page *80*. | |

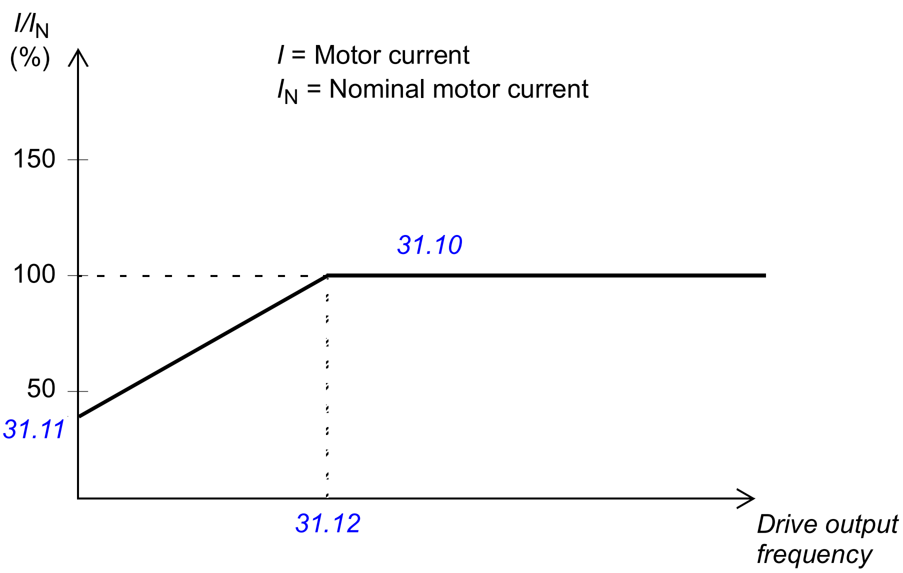
| Bit | Function |
|---|---|
| 0 | Ena sup (Enable supervision)<br>0 = Disabled: Supervision disabled.<br>1 = Enabled: Supervision enabled. |
| 1 | Ena warn (Enable warning)<br>0 = Disabled<br>1 = Enabled: Drive generates alarm *STALL (0x7121)* upon a stall condition. |
| 2 | Ena fault (Enable fault)<br>0 = Disabled<br>1 = Enabled: Drive trips on fault *STALL (0x7121)* upon a stall condition. |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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*.<br>**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 |

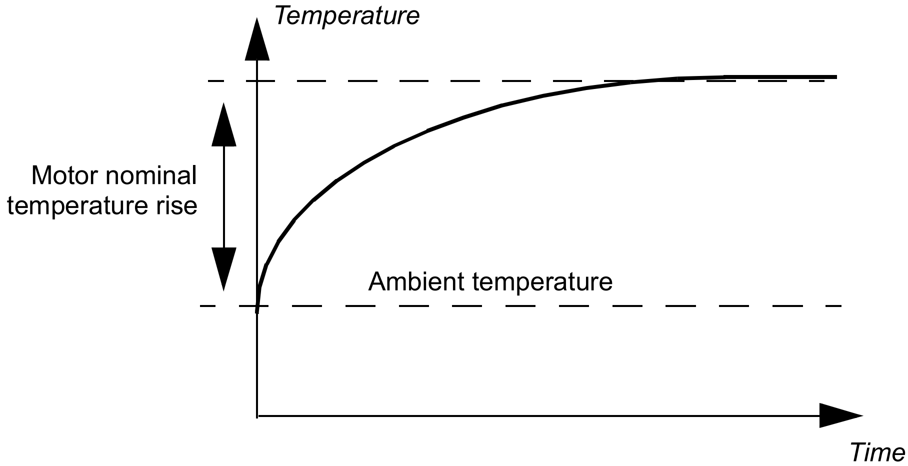
| | | | |
|---|---|---|---|
| **31 Motor therm prot** | | Motor temperature measurement and thermal protection settings. | |
| 31.01 | Mot temp1 prot | Selects how the drive reacts when motor overtemperature is detected by motor thermal protection 1. | |
| | No | Motor thermal protection 1 inactive. | 0 |
| | Alarm | The drive generates alarm *MOTOR TEMPERATURE (0x4310)* if the temperature exceeds the alarm level defined by parameter *31.03 Mot temp1 almLim*. | 1 |
| | Fault | The drive generates alarm *MOTOR TEMPERATURE (0x4310)* or trips on fault *MOTOR OVERTEMP (0x4310)* if the temperature exceeds the alarm/fault level defined by parameter *31.02 Mot temp1 almLim* / *31.03 Mot temp1 almLim* (whichever is lower). | 2 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 *31.01 Mot temp1 prot*. | |
| | Estimated | 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.<br>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).<br>**WARNING!** The model does not protect the motor if it does not cool properly due to dust and dirt. | 0 |
| | PTC JCU | The temperature is supervised using 1…3 PTC sensors connected to digital input DI5. | 4 |
| | 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. | 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. | 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. | 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. | 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. | 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. | 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 *Alarm* or *Fault*). | |
| | 0 … 200 °C | Motor overtemperature alarm limit. | 1 = 1 °C |
| 31.04 | Mot temp1 fltLim | Defines the fault limit for the motor thermal protection 1 (when parameter *31.01 Mot temp1 prot* is set to *Fault*). | |
| | 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 (0x4313)* when the temperature exceeds the alarm level defined by parameter *31.07 Mot temp2 almLim*. | 1 |
| | Fault | The drive generates alarm *MOTTEMPAL2 (0x4313)* or trips on fault *MOTOR TEMP2 (0x4313)* when the temperature exceeds the alarm/fault level defined by parameter *31.07 Mot temp2 almLim* / *31.08 Mot temp2 fltLim* (whichever is lower). | 2 |

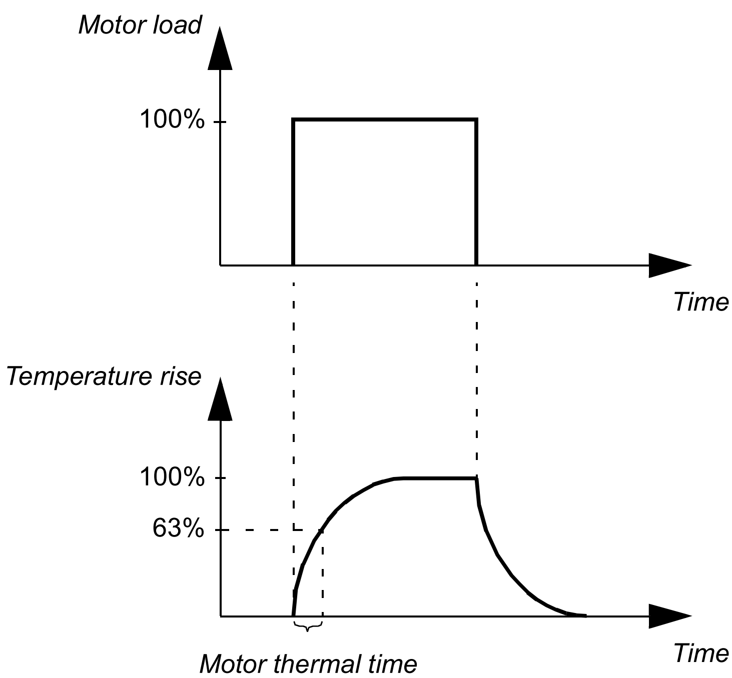
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 31.06 | Mot temp2 src | Selects the means of temperature measurement for motor thermal protection 2. When overtemperature is detected the drive reacts as defined by parameter *31.05 Mot temp2 prot*. | |
| | Estimated | The temperature is supervised 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.<br>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).<br>**WARNING!** The model does not protect the motor if it does not cool properly due to dust and dirt. | 0 |
| | PTC JCU | The temperature is supervised using 1…3 PTC sensors connected to digital input DI5. | 4 |
| | 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. | 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. | 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. | 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. | 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. | 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. | 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 *Alarm* or *Fault*). | |
| | 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 *Fault*). | |
| | 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 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 31.10 | Mot load curve | Defines the load curve together with parameters *31.11 Zero speed load* and *31.12 Break point*<br>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.<br>The load curve is used by the motor thermal protection model when parameter *31.02 Mot temp1 src* is set to *Estimated*.<br><br> | |
| | 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.<br>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*.<br>See parameter *31.10 Mot load curve*. | |
| | 0.01 … 500.00 Hz | Break point for the motor load curve. | 100 = 1 Hz |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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.<br>The temperature rise value is used by the motor thermal protection model when parameter *31.02 Mot temp1 src* is set to *Estimated*. | |




| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 0 … 300 °C | Temperature rise. | 1 = 1 °C |
| 31.14 | Mot therm time | 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.<br>The motor thermal protection model is used when parameter *31.02 Mot temp1 src* is set to *Estimated*. | |




| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 100 … 10000 s | Motor thermal time constant. | 1 = 1 s |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| **32 Automatic reset** | | Configuration of conditions for automatic fault resets. | |
| 32.01 | Autoreset sel | 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.<br>The bits of the binary number correspond to the following faults: | |

| Bit | Fault |
|---|---|
| 0 | AR overcurrent |
| 1 | AR overvoltage |
| 2 | AR undervoltage |
| 3 | AR AI min |
| 4 | Reserved |
| 5 | AR external |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 |
| 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.<br>See also section *Signal supervision* on page *81*. | |
| 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. | 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. | 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. | 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. | 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 *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | Frequency | *01.03 Output frequency* (see page *112*). | 1073742083 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Current | *01.04 Motor current* (see page *112*). | 1073742084 |
| | Current % | *01.05 Motor current %* (see page *112*). | 1073742085 |
| | Torque | *01.06 Motor torque* (see page *112*). | 1073742086 |
| | Dc-voltage | *01.07 Dc-voltage* (see page *112*). | 1073742087 |
| | Power inu | *01.22 Power inu out* (see page *112*). | 1073742102 |
| | Power motor | *01.23 Motor power* (see page *112*). | 1073742103 |
| | SpRef unramp | *03.03 SpeedRef unramp* (see page *123*). | 1073742595 |
| | SpRef ramped | *03.05 SpeedRef ramped* (see page *123*). | 1073742597 |
| | SpRef used | *03.06 SpeedRef used* (see page *123*). | 1073742598 |
| | TorqRef used | *03.14 Torq ref used* (see page *123*). | 1073742606 |
| | Process act | *04.01 Act val* (see page *123*). | 1073742849 |
| | Proc PID out | *04.05 Process PID out* (see page *123*). | 1073742853 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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. | 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. | 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. | 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. | 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 *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | Frequency | *01.03 Output frequency* (see page *112*). | 1073742083 |
| | Current | *01.04 Motor current* (see page *112*). | 1073742084 |
| | Current % | *01.05 Motor current %* (see page *112*). | 1073742085 |
| | Torque | *01.06 Motor torque* (see page *112*). | 1073742086 |
| | Dc-voltage | *01.07 Dc-voltage* (see page *112*). | 1073742087 |
| | Power inu | *01.22 Power inu out* (see page *112*). | 1073742102 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Power motor | *01.23 Motor power* (see page *112*). | 1073742103 |
| | SpRef unramp | *03.03 SpeedRef unramp* (see page *123*). | 1073742595 |
| | SpRef ramped | *03.05 SpeedRef ramped* (see page *123*). | 1073742597 |
| | SpRef used | *03.06 SpeedRef used* (see page *123*). | 1073742598 |
| | TorqRef used | *03.14 Torq ref used* (see page *123*). | 1073742606 |
| | Process act | *04.01 Act val* (see page *123*). | 1073742849 |
| | Proc PID out | *04.05 Process PID out* (see page *123*). | 1073742853 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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. | 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. | 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. | 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. | 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 *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | Frequency | *01.03 Output frequency* (see page *112*). | 1073742083 |
| | Current | *01.04 Motor current* (see page *112*). | 1073742084 |
| | Current % | *01.05 Motor current %* (see page *112*). | 1073742085 |
| | Torque | *01.06 Motor torque* (see page *112*). | 1073742086 |
| | Dc-voltage | *01.07 Dc-voltage* (see page *112*). | 1073742087 |
| | Power inu | *01.22 Power inu out* (see page *112*). | 1073742102 |
| | Power motor | *01.23 Motor power* (see page *112*). | 1073742103 |
| | SpRef unramp | *03.03 SpeedRef unramp* (see page *123*). | 1073742595 |
| | SpRef ramped | *03.05 SpeedRef ramped* (see page *123*). | 1073742597 |
| | SpRef used | *03.06 SpeedRef used* (see page *123*). | 1073742598 |
| | TorqRef used | *03.14 Torq ref used* (see page *123*). | 1073742606 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Process act | *04.01 Act val* (see page *123*). | 1073742849 |
| | Proc PID out | *04.05 Process PID out* (see page *123*). | 1073742853 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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** | | Configuration of user load curve.<br>See also section *User-definable load curve* on page *71*. | |
| 34.01 | Overload func | Configures the supervision of the upper boundary of the user load curve. | |

| Bit | Function |
|---|---|
| 0 | Ena sup (Enable supervision)<br>0 = Disabled: Supervision disabled.<br>1 = Enabled: Supervision enabled. |
| 1 | Input value sel (Input value selection)<br>0 = Current: Current is supervised.<br>1 = Torque: Torque is supervised. |
| 2 | Ena warn (Enable warning)<br>0 = Disabled<br>1 = Enabled: Drive generates alarm *LCURVE (0x2312)* when the curve is exceeded. |
| 3 | Ena fault (Enable fault)<br>0 = Disabled<br>1 = Enabled: Drive trips on fault *LOAD CURVE (0x2312)* when the curve is exceeded. |
| 4 | Ena lim integ (Enable limit integration)<br>0 = Disabled<br>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)<br>0 = Disabled<br>1 = Enabled: The current or torque is always limited by the upper boundary of the load curve. |

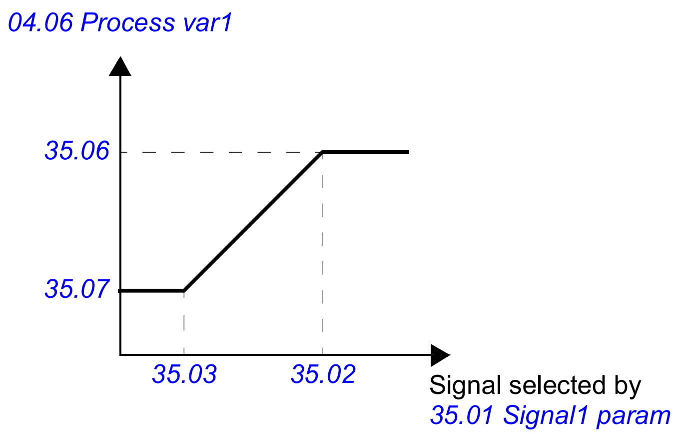
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 34.02 | Underload func | Configures the supervision of the lower boundary of the user load curve. | |

| Bit | Function |
|---|---|
| 0 | Ena sup (Enable supervision)<br>0 = Disabled: Supervision disabled.<br>1 = Enabled: Supervision enabled. |
| 1 | Input value sel (Input value selection)<br>0 = Current: Current is supervised.<br>1 = Torque: Torque is supervised. |
| 2 | Ena warn (Enable warning)<br>0 = Disabled<br>1 = Enabled: Drive generates alarm *LCURVE (0x2312)* when the load remains below the curve for longer than the time defined by parameter *34.20 Underload time*. |
| 3 | Ena fault (Enable fault)<br>0 = Disabled<br>1 = Enabled: Drive trips on fault *LOAD CURVE (0x2312)* when the load remains below the curve for longer than the time defined by parameter *34.20 Underload time*. |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 |

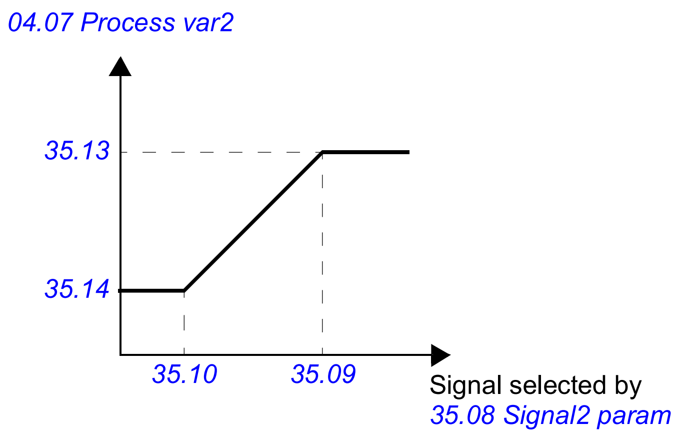
| | | | |
|---|---|---|---|
| **35 Process variable** | | Selection and modification of process variables for display as parameters *04.06* … *04.08*. | |
| 35.01 | Signal1 param | Selects a signal to be provided as parameter *04.06 Process var1*. | |
| | Speed rpm | *01.01 Motor speed rpm* (see page *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | Frequency | *01.03 Output frequency* (see page *112*). | 1073742083 |
| | Current | *01.04 Motor current* (see page *112*). | 1073742084 |
| | Current % | *01.05 Motor current %* (see page *112*). | 1073742085 |
| | Torque | *01.06 Motor torque* (see page *112*). | 1073742086 |
| | Dc-voltage | *01.07 Dc-voltage* (see page *112*). | 1073742087 |
| | Power inu | *01.22 Power inu out* (see page *112*). | 1073742102 |
| | Power motor | *01.23 Motor power* (see page *112*). | 1073742103 |
| | SpRef unramp | *03.03 SpeedRef unramp* (see page *123*). | 1073742595 |
| | SpRef ramped | *03.05 SpeedRef ramped* (see page *123*). | 1073742597 |
| | SpRef used | *03.06 SpeedRef used* (see page *123*). | 1073742598 |
| | TorqRef used | *03.14 Torq ref used* (see page *123*). | 1073742606 |
| | Process act | *04.01 Act val* (see page *123*). | 1073742849 |
| | Proc PID out | *04.05 Process PID out* (see page *123*). | 1073742853 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |

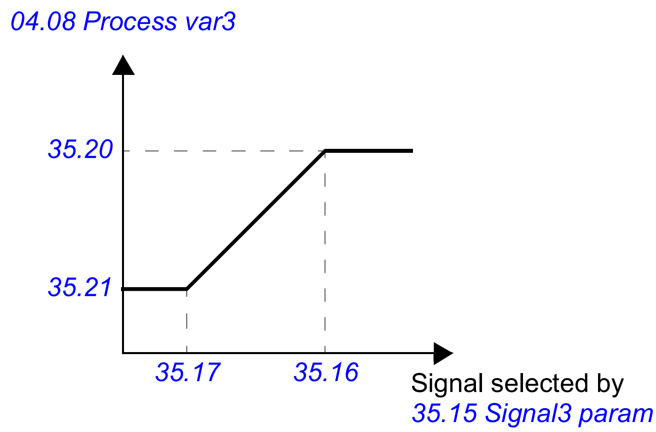
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 35.02 | Signal1 max | Defines the real value of the selected signal that corresponds to the maximum display value defined by parameter *35.06 Proc var1 max*.<br><br> | |
| | -32768…32768 | Real signal value corresponding to maximum process variable 1 value. | 1 = 1 |
| 35.03 | Signal1 min | 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*. | |
| | -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 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 12 | mV | 12 |
| | 13 | kW | 13 |
| | 14 | W | 14 |
| | 15 | kWh | 15 |
| | 16 | F | 16 |
| | 17 | hp | 17 |
| | 18 | MWh | 18 |
| | 19 | m/s | 19 |
| | 20 | m3/h | 20 |
| | 21 | dm3/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 | m3/s | 39 |
| | 40 | m3/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 | ft3/s | 50 |
| | 51 | ft3/m | 51 |
| | 52 | ft3/h | 52 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 53 | lb/s | 53 |
| | 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 | 69 |
| | 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 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 96 | mOhm | 96 |
| | 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 *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | Frequency | *01.03 Output frequency* (see page *112*). | 1073742083 |
| | Current | *01.04 Motor current* (see page *112*). | 1073742084 |
| | Current % | *01.05 Motor current %* (see page *112*). | 1073742085 |
| | Torque | *01.06 Motor torque* (see page *112*). | 1073742086 |
| | Dc-voltage | *01.07 Dc-voltage* (see page *112*). | 1073742087 |
| | Power inu | *01.22 Power inu out* (see page *112*). | 1073742102 |
| | Power motor | *01.23 Motor power* (see page *112*). | 1073742103 |
| | SpRef unramp | *03.03 SpeedRef unramp* (see page *123*). | 1073742595 |
| | SpRef ramped | *03.05 SpeedRef ramped* (see page *123*). | 1073742597 |
| | SpRef used | *03.06 SpeedRef used* (see page *123*). | 1073742598 |
| | TorqRef used | *03.14 Torq ref used* (see page *123*). | 1073742606 |
| | Process act | *04.01 Act val* (see page *123*). | 1073742849 |
| | Proc PID out | *04.05 Process PID out* (see page *123*). | 1073742853 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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*.<br> | |
| | -32768…32768 | Real signal value corresponding to maximum process variable 2 value. | 1 = 1 |
| 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 *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | Frequency | *01.03 Output frequency* (see page *112*). | 1073742083 |
| | Current | *01.04 Motor current* (see page *112*). | 1073742084 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Current % | *01.05 Motor current %* (see page *112*). | 1073742085 |
| | Torque | *01.06 Motor torque* (see page *112*). | 1073742086 |
| | Dc-voltage | *01.07 Dc-voltage* (see page *112*). | 1073742087 |
| | Power inu | *01.22 Power inu out* (see page *112*). | 1073742102 |
| | Power motor | *01.23 Motor power* (see page *112*). | 1073742103 |
| | SpRef unramp | *03.03 SpeedRef unramp* (see page *123*). | 1073742595 |
| | SpRef ramped | *03.05 SpeedRef ramped* (see page *123*). | 1073742597 |
| | SpRef used | *03.06 SpeedRef used* (see page *123*). | 1073742598 |
| | TorqRef used | *03.14 Torq ref used* (see page *123*). | 1073742606 |
| | Process act | *04.01 Act val* (see page *123*). | 1073742849 |
| | Proc PID out | *04.05 Process PID out* (see page *123*). | 1073742853 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 35.16 | Signal3 max | Defines the real value of the selected signal that corresponds to the maximum display value defined by parameter *35.20 Proc var3 max*.<br><br> | |
| | -32768…32768 | Real signal value corresponding to maximum process variable 3 value. | 1 = 1 |
| 35.17 | Signal3 min | 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*. | |
| | -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 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 35.19 | Proc var3 unit | Specifies the unit for parameter *04.08 Process var3* (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.<br>See also section *Timers* on page *74*. | |
|---|---|---|---|
| 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 |
| | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | 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. | |

| Bit | Function |
|---|---|
| 0 | Timer1 mode<br>0 = Daily<br>1 = Weekly |
| 1 | Timer2 mode<br>0 = Daily<br>1 = Weekly |
| 2 | Timer3 mode<br>0 = Daily<br>1 = Weekly |
| 3 | Timer4 mode<br>0 = Daily<br>1 = Weekly |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 36.03 | Start time1 | Defines the start time for time period 1. | |
| | 00:00:00 …<br>24:00:00 | Start time for time period 1. | 1 = 1 s<br>(24:00:00 = 86400) |
| 36.04 | Stop time1 | Defines the stop time for time period 1. | |
| | 00:00:00 …<br>24:00:00 | Stop time for time period 1. | 1 = 1 s<br>(24:00:00 = 86400) |
| 36.05 | Start day1 | Defines the week day on which time period 1 begins. | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 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 |
| | 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) |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 |
| | 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 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Sunday | Time period 1 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | 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 | Selects which time periods (1…4) are used with timed function 1. Also determines whether boost is used with timed function 1.<br>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.<br>The bits of the binary number correspond to the following functions: | |

| 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 |
|---|---|---|---|
| 36.22 | Timed func2 | Selects which time periods (1…4) are used with timed function 2. Also determines whether boost is used with timed function 2.<br>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.<br>The bits of the binary number correspond to the following functions: | |

| 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 | Selects which time periods (1…4) are used with timed function 3. Also determines whether boost is used with timed function 3.<br>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.<br>The bits of the binary number correspond to the following functions: | |

| 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) |

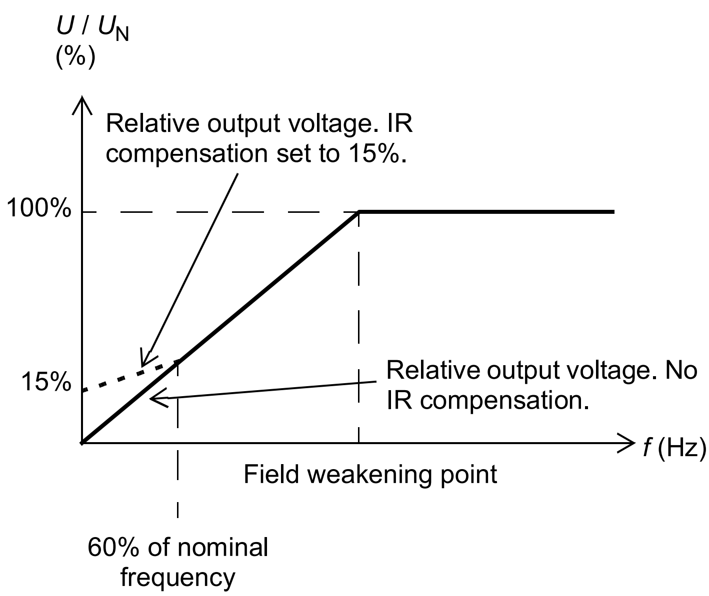
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 36.24 | Timed func4 | Selects which time periods (1…4) are used with timed function 4. Also determines whether boost is used with timed function 4.<br>The parameter is a 16-bit word with each bit corresponding to a function. Whenever a bit is set to 1, the corresponding function is in use.<br>The bits of the binary number correspond to the following functions: | |

| 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 |
|---|---|---|---|
| **38 Flux ref** | | Flux reference and *U/f* curve settings.<br>See also section *User-definable U/f curve* on page 72. | |
| 38.01 | Flux ref | Sets the flux reference (in percent of parameter *99.08 Mot nom freq*) at field weakening point. | |
| | 0 … 200% | Flux reference at field weakening point. | 1 = 1% |
| 38.03 | U/f curve func | Selects the form of the *U/f* (voltage/frequency) curve below the field weakening point.<br>**Note:** This functionality can be used in scalar control only, i.e. when *99.05 Motor ctrl mode* is set to *Scalar*. | |
| | Linear | Linear *U/f* curve. Recommended for constant-torque applications. | 0 |
| | Quadratic | Quadratic *U/f* curve. Recommended for centrifugal pump and fan applications. | 1 |
| | User | Custom *U/f* curve. The curve is formed by the points defined by parameters *38.04*…*38.13*. | 2 |
| 38.04 | U/f curve freq1 | Defines the frequency at the 1st point on the custom *U/f* curve in percent of parameter *99.08 Mot nom freq*. | |
| | 1 … 500% | 1st point, frequency. | 1 = 1% |
| 38.05 | U/f curve freq2 | Defines the frequency at the 2nd point on the custom *U/f* curve in percent of parameter *99.08 Mot nom freq*. | |

| 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 *U/f* 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 *U/f* 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 *U/f* 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 *U/f* 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 *U/f* 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 *U/f* 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 *U/f* 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 *U/f* 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 *109*). | - |

| | | | |
|---|---|---|---|
| **40 Motor control** | | Motor control settings such as performance/noise optimization, slip gain, voltage reserve and IR compensation. | |
| 40.01 | Motor noise | An optimization setting for balancing between control performance and motor noise level. | |
| | Cyclic | Control performance optimized for cyclic load applications.<br>**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.<br>**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 | 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.<br>**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%). | |
| | 0 … 200% | Slip gain. | 1 = 1% |
| 40.04 | Voltage reserve | Defines the minimum allowed voltage reserve. When the voltage reserve has decreased to the set value, the drive enters the field weakening area.<br>If the intermediate circuit DC voltage $U_{dc}$ = 550 V and the voltage reserve is 5%, the RMS value of the maximum output voltage in steady-state operation is<br>0.95 × 550 V / sqrt(2) = 369 V<br>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. | |
| | -4 … 50% | Voltage reserve. | 1 = 1% |
| 40.07 | IR-compensation | 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.<br><br><br>See also section *IR compensation for a scalar controlled drive* on page *70*. | |
| | 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 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 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 |
| **44 Maintenance** | | Maintenance counter configuration.<br>See also section *Maintenance counters* on page *82*. | |
| 44.01 | Ontime1 func | 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.<br>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. | |

| Bit | Function |
|---|---|
| 0 | Counter mode<br>0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds.<br>1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset. |
| 1 | Alarm ena (Alarm enable)<br>0 = Disable: No alarm is given when limit is reached.<br>1 = Enable: Alarm is given when limit is reached. |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 *126*). | 1073939969 |
| | Charged | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | 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 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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.<br>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. | |

| Bit | Function |
|---|---|
| 0 | Counter mode<br>0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds.<br>1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset. |
| 1 | Alarm ena (Alarm enable)<br>0 = Disable: No alarm is given when limit is reached.<br>1 = Enable: Alarm is given when limit is reached. |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 *126*). | 1073939969 |
| | Charged | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | 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 | 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.<br>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. | |

| Bit | Function |
|---|---|
| 0 | Counter mode<br>0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds.<br>1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset. |
| 1 | Alarm ena (Alarm enable)<br>0 = Disable: No alarm is given when limit is reached.<br>1 = Enable: Alarm is given when limit is reached. |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 *126*). | 1073939969 |
| | Charged | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | 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 | 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.<br>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. | |

| Bit | Function |
|---|---|
| 0 | Counter mode<br>0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds.<br>1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset. |
| 1 | Alarm ena (Alarm enable)<br>0 = Disable: No alarm is given when limit is reached.<br>1 = Enable: Alarm is given when limit is reached. |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 *126*). | 1073939969 |
| | Charged | Bit 9 of *06.02 Status word2* (see page *127*). | 1074333186 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 44.16 | Edge count2 lim | Sets the alarm limit for rising edge counter 1. 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 | 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).<br>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.<br>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. | |

| Bit | Function |
|---|---|
| 0 | Counter mode<br>0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds.<br>1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset. |
| 1 | Alarm ena (Alarm enable)<br>0 = Disable: No alarm is given when limit is reached.<br>1 = Enable: Alarm is given when limit is reached. |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 *112*). | 1073742081 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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 | 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).<br>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.<br>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. | |

| Bit | Function |
|---|---|
| 0 | Counter mode<br>0 = Loop: If alarm is enabled by bit 1, the alarm stays active only for 10 seconds.<br>1 = Saturate: If alarm is enabled by bit 1, the alarm stays active until reset. |
| 1 | Alarm ena (Alarm enable)<br>0 = Disable: No alarm is given when limit is reached.<br>1 = Enable: Alarm is given when limit is reached. |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 *112*). | 1073742081 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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 *112*). When the signal reaches the limit, alarm *2056 COOLING FAN* 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 *112*). 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 *112*). When the signal reaches the limit, the alarm specified by parameter *44.33 kWh inv alm sel* is given. | |
| | 0…2147483647 kWh | 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.<br>See also section *Energy savings calculator* on page *81*. | |
|---|---|---|---|
| 45.01 | Energy optim | Enables/disables the energy optimization function. The function optimizes the flux so that the 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. | |
| | 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 $CO_2$ 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 in carbon dioxide emissions in metric tons).<br>*01.37 Saved CO2* = *01.35 Saved energy* (MWh) × *45.07 CO2 Conv factor* (tn/MWh). | |
| | 0.0 … 10.0 | Conversion factor for converting energy into $CO_2$ emissions (kg/kWh or tn/MWh). | 1 = 1 |
| 45.08 | Pump ref power | Pump 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*. | |
| | 0.0 … 1000.0% | Pump 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.<br>See also *DC voltage control* on page *76*. | |
| 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. | |
| | 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. | |
| | Disable | Undervoltage control disabled. | 0 |
| | Enable | Undervoltage control enabled. | 1 |
| 47.03 | SupplyVoltAutoId | 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 SupplyVoltAutoId*. | |
| | 0 … 1000 V | Nominal supply voltage. | 10 = 1 V |
| **49 Data storage** | | Data storage parameters reserved for the user.<br>These parameters can be written to and read from using other parameters' pointer settings. Four 16-bit and four 32-bit storage parameters are available. | |
| 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.<br>See also chapter *Control through a fieldbus adapter* on page *357*. | |
| 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 *50.03 Comm loss t out*. | |
| | No | Communication break detection disabled. | 0 |
| | Fault | Communication break detection active. Upon a communication break, the drive trips on fault *FIELDBUS COMM (0x7510)* and coasts to stop. | 1 |
| | Spd ref Safe | Communication break detection active. Upon a communication break, the drive generates alarm *FIELDBUS COMM (0x7510)* and sets the speed to the value defined by parameter *30.02 Speed ref safe*.<br>**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 (0x7510)* 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.<br>**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 *50.02 Comm loss func* 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 | FBA 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 *50.06 FBA act1 tr src*. | 0 |
| | 1 | Reserved. | 1 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Speed | Fieldbus adapter module 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 User's Manual of the appropriate fieldbus adapter module. | 2 |
| 50.05 | FBA ref2 modesel | Selects the fieldbus reference FBA REF2 scaling.<br>See parameter *50.04 FBA ref1 modesel*. | |
| | Raw data | See parameter *50.04 FBA ref1 modesel*. | 0 |
| | 1 | Reserved. | 1 |
| | Speed | See parameter *50.04 FBA ref1 modesel*. | 2 |
| 50.06 | FBA act1 tr src | Selects the source for fieldbus actual value 1 when parameter *50.04 FBA ref1 modesel* / *50.05 FBA ref2 modesel* is set to *Raw data*. | |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 50.07 | FBA act2 tr src | Selects the source for fieldbus actual value 2 when parameter *50.04 FBA ref1 modesel* / *50.05 FBA ref2 modesel* is set to *Raw data*. | |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 50.08 | FBA sw bit12 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<br>Pointer | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 50.09 | FBA sw bit13 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<br>Pointer | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 50.10 | FBA sw bit14 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<br>Pointer | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 50.11 | FBA sw bit15 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<br>Pointer | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 50.15 | Fb cw used | Selects the fieldbus Control Word which controls the drive. | |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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). | |

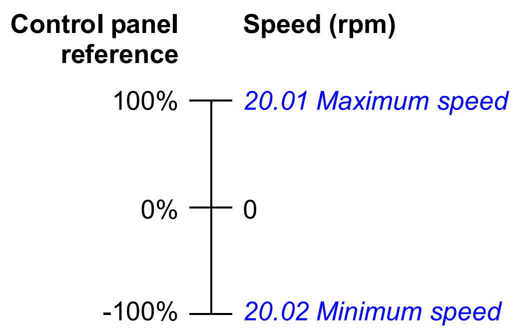
| Bit | Name | Information |
|---|---|---|
| 0 | Run enable func | 1 = Parameter only:<br>Drive writes value 1 to the bit when the external run enable signal (par. *10.11 Run enable*) has value 1. |
| | | 0 = Param AND Fb cw:<br>Drive writes value 1 to the bit when the external run enable signal (par. *10.11 Run enable*) is 1 AND *02.22 FBA main cw* bit 7 (Run enable) is 1. |

| | | | |
|---|---|---|---|
| **51 FBA settings** | | Fieldbus adapter-specific settings. | |
| 51.01 | FBA type | Displays the type of the connected fieldbus adapter module.<br>0 = Fieldbus module is not found, or it is not properly connected, or parameter *50.01 FBA enable* is set to *Disable*,<br>1 = PROFIBUS DP, 32 = CANopen, 37 = DeviceNet,<br>128 = Ethernet, 132 = PROFINET IO, 135 = EtherCAT,<br>136 = Ethernet POWERLINK, 485 = RS-485,<br>62944 = SERCOS interface. | |
| 51.02 | FBA par2 | Parameters *51.02…51.26* are adapter module-specific. For more information, see the User's Manual 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 *Done*.<br>**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.<br>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.<br>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.<br>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.<br>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*. | |

| 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 control panel and DriveStudio PC tool. Also determines the unit of signal *02.34 Panel ref*.<br>**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:<br> | 1 |
| **58 Embedded Modbus** | | Configuration parameters for the embedded fieldbus (EFB) interface.<br>See also chapter *Control through the embedded fieldbus interface* on page *329*. | |
| 58.01 | Protocol ena sel | Enables/disables the embedded fieldbus communication protocol.<br>**Note:** When the embedded fieldbus interface is enabled, the drive-to-drive link operation (parameter group 76) 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 |
| | 57600 | 57.6 kbit/s. | 4 |
| | 76800 | 76.8 kbit/s. | 5 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 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:<br>Comm loss timeout × 100 ms<br>**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 fault *EFB COMM LOSS (0x7540)*. | 1 |
| | Safe speed | Drive generates alarm *EFB COMM LOSS (0x7540)* and takes the safe speed into use (see parameter *30.02 Speed ref safe*). | 2 |
| | Last speed | Drive generates alarm *EFB COMM LOSS (0x7540)* 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 *DCU 16-bit profile* on page 348. | |
| | 1…65535 | Scaling factor. | 1 = 1 |
| 58.15 | Comm diagnostics | 16-bit packed boolean data word for the communication diagnostics flag bits. Read only. | |

| Bit | Information |
|---|---|
| 0 | Reserved. |
| 1 | Last received packet was not for this node. |
| 2 | Reserved. |
| 3 | At least one packet has been successfully received after the power up. |
| 4 | Reserved. |
| 5 | Communication time-out has occured. |
| 6…7 | Reserved. |
| 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…14 | Reserved. |
| 15 | Last write was not successful because only one register was used to read a 32-bit value. |
| 16…31 | Reserved. |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 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 |
| 58.20 | CRC errors | Shows the number of messages with Cyclic Redundancy Check (CRC) errors which the drive has received. Read only.<br>**Note:** High electromagnetic noise levels may generate errors. | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 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).<br>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. | |
| | 0…9999 | Parameter address. Format: xxyy, where:<br>xx = parameter group<br>yy = parameter index | 1 = 1 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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.<br>See also section *Load analyzer* on page *82*. | |
| 64.01 | PVL signal | Selects the signal to be monitored by the peak value logger.<br>The signal is filtered using the filtering time specified by parameter *64.02 PVL filt time*.<br>The peak value is stored, along with other pre-selected signals at the time, into parameters *64.06*…*64.11*.<br>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 *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | Frequency | *01.03 Output frequency* (see page *112*). | 1073742083 |
| | Current | *01.04 Motor current* (see page *112*). | 1073742084 |
| | Current % | *01.05 Motor current %* (see page *112*). | 1073742085 |
| | Torque | *01.06 Motor torque* (see page *112*). | 1073742086 |
| | Dc-voltage | *01.07 Dc-voltage* (see page *112*). | 1073742087 |
| | Power inu | *01.22 Power inu out* (see page *112*). | 1073742102 |
| | Power motor | *01.23 Motor power* (see page *112*). | 1073742103 |
| | Process act | *04.01 Act val* (see page *123*). | 1073742849 |
| | Proc PID out | *04.05 Process PID out* (see page *123*). | 1073742853 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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 *109*). | - |
| | Pointer | | |

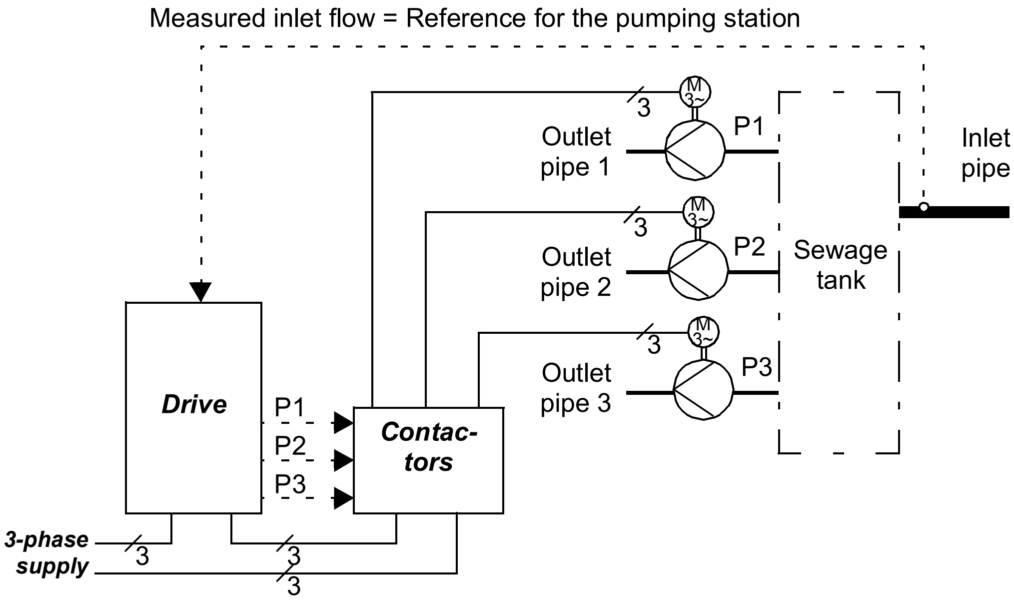
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 64.04 | AL signal | Selects the signal to be monitored by amplitude logger 2. The signal is sampled at 200 ms intervals when the drive is running.<br>The results are displayed by parameters *64.24*…*64.33*. Each parameter represents an amplitude range, and shows what portion of the samples fall within that range.<br>The signal value corresponding to 100% is defined by parameter *64.05 AL signal base*.<br>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*.<br>**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 *Hardware Manual)*. | |
| | Speed rpm | *01.01 Motor speed rpm* (see page *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | Frequency | *01.03 Output frequency* (see page *112*). | 1073742083 |
| | Current | *01.04 Motor current* (see page *112*). | 1073742084 |
| | Current % | *01.05 Motor current %* (see page *112*). | 1073742085 |
| | Torque | *01.06 Motor torque* (see page *112*). | 1073742086 |
| | Dc-voltage | *01.07 Dc-voltage* (see page *112*). | 1073742087 |
| | Power inu | *01.22 Power inu out* (see page *112*). | 1073742102 |
| | Power motor | *01.23 Motor power* (see page *112*). | 1073742103 |
| | Process act | *04.01 Act val* (see page *123*). | 1073742849 |
| | Proc PID out | *04.05 Process PID out* (see page *123*). | 1073742853 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 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 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 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 |
| 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% |

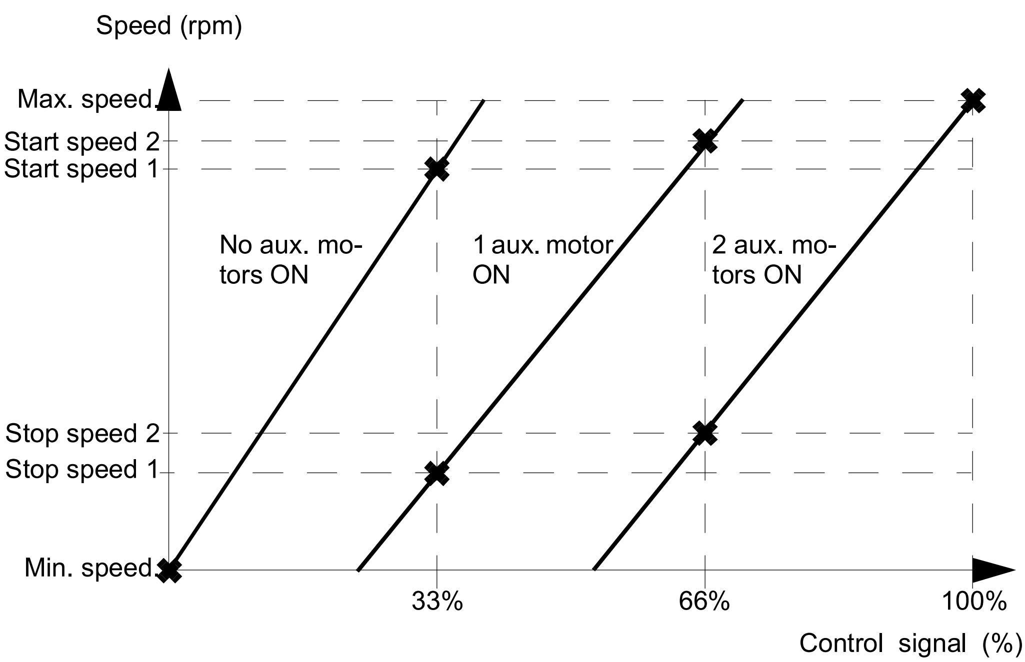
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 64.26 | AL2 20 to 30% | Percentage of samples recorded by amplitude logger 2 that fall between 20 and 30%. | |
| | 0.00 … 100.00% | Amplitude logger 2 samples between 20 and 30%. | 100 = 1% |
| 64.27 | AL2 30 to 40% | Percentage of samples 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% |

| | | | |
|---|---|---|---|
| **75 Pump logic** | | Configuration settings for the pump station. | |
| 75.01 | Operation mode | Selects the pump control mode. | |
| | Off | Use this setting for a single pump. | 0 |
| | Trad ctrl | Traditional pump control mode.<br>One pump at a time is controlled by the drive. The remaining pumps are direct-on-line pumps that are started and stopped by the drive logic. | 1 |

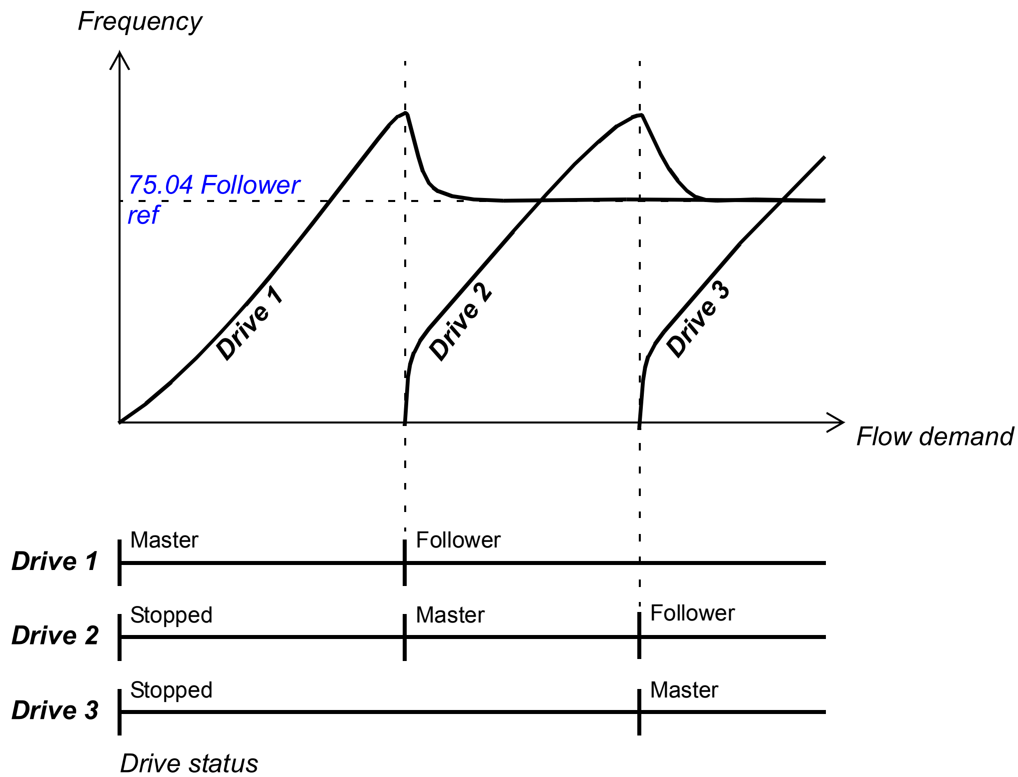
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Reg bypass | PID controller bypass mode.<br>The signal selected by parameters 28.01…28.04 is used as the reference. The automatic starting and stopping of direct-on-line pumps is related to this actual value instead of the output of the PID controller.<br>This setting can be used in applications with a low number of sensors and low accuracy requirements.<br>*Example:* The capacity of the pumping station (outlet flow) follows the measured inlet flow. | 2 |




In the diagram below, the slopes of the lines describe the relation between the control signal (selected by parameters 28.01…28.04) and the speed of the controlled pump in a three-motor system. At full control signal level, all pumps are operating at maximum speed.

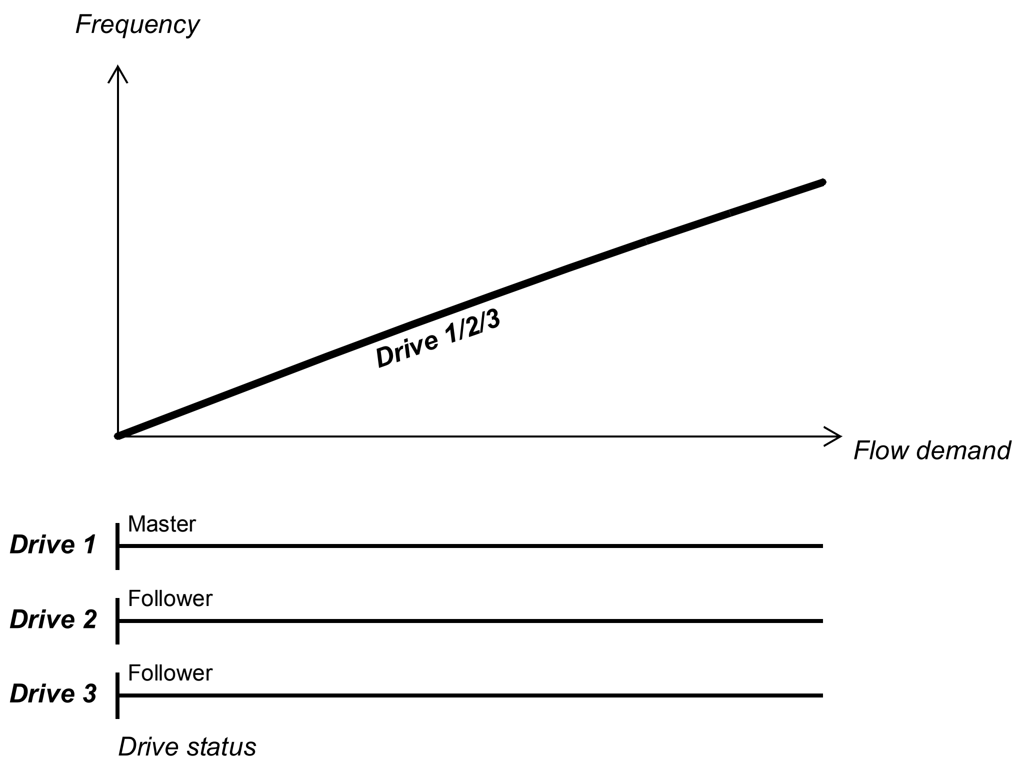


| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Multipump | Multiple drives, each controlling a separate pump, are connected together using the drive-to-drive link. | 3 |
| 75.02 | Nbr of pumps | Total number of pumps used in the application, including the pump connected directly to the drive. | |
| | 0…8 | Number of pumps. | 1 = 1 |
| 75.03 | Follower mode | Selects the source of reference when the drive is a follower. | |
| | Const speed | Follower drives are started and stopped by the control logic in the master drive. The master receives its reference from the PI controller.<br>When flow demand increases, new pumps are started.<br>If parameter *76.10 Master location* is set to *In start*, the latest drive to start becomes the master; at the same time, the previously-started drive becomes a follower and starts to follow the reference defined by parameter *75.04 Follower ref*.<br>If parameter *76.10 Master location* is set to *Stable*, the drive that was started first remains the master. | 0 |

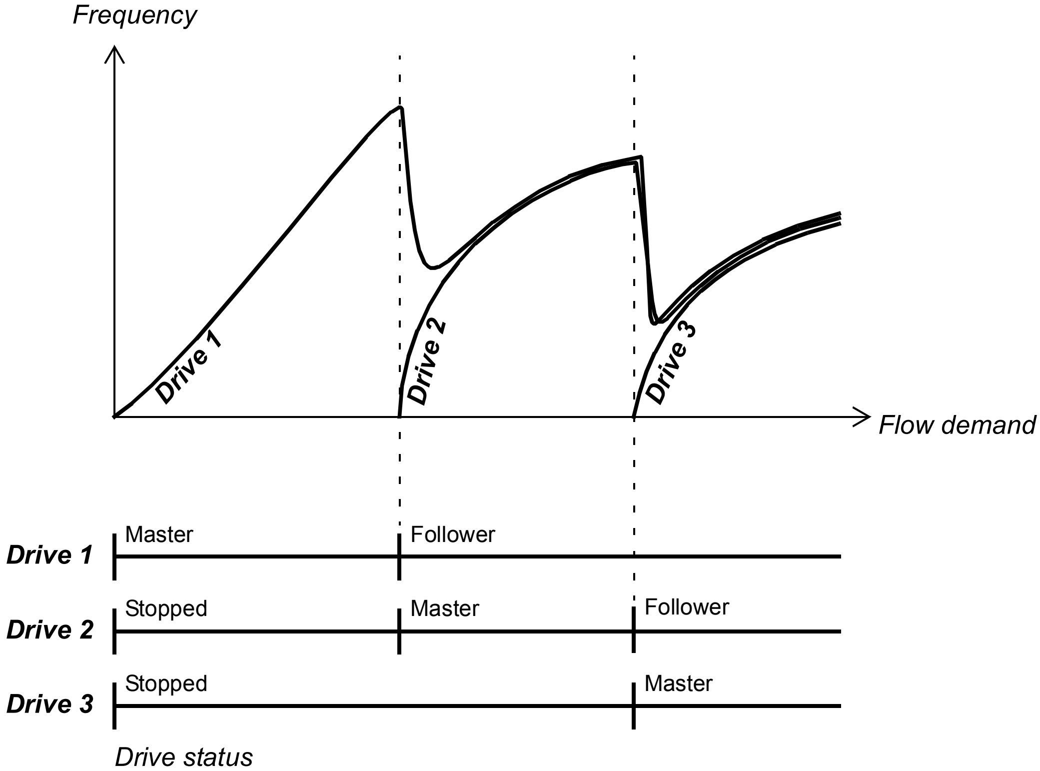



See also the diagrams at parameter *75.04 Follower ref*.

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Copy of mstr | The drive follows the same start/stop commands and reference (received from the PI controller) as the master.<br>With this setting, the drive does not become master when started.<br>In the example shown, drive 1 is master; drives 2 and 3 have parameter *75.03 Follower mode* set to *Copy of mstr*. | 1 |

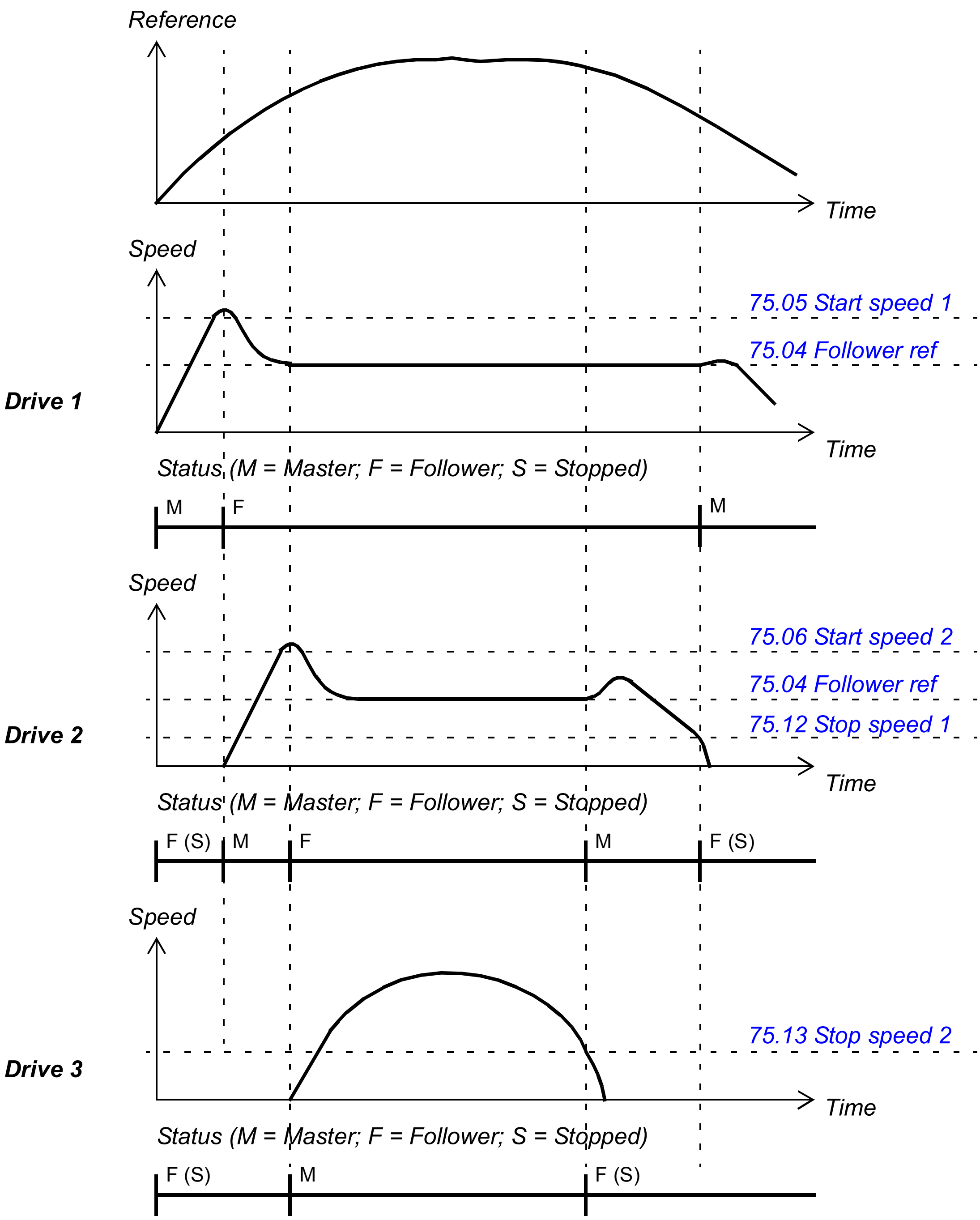


| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Master speed | The drive follows the same reference (received from the PI controller) as the master, but is started and stopped by the logic. This is usually the most economical follower mode. | 2 |



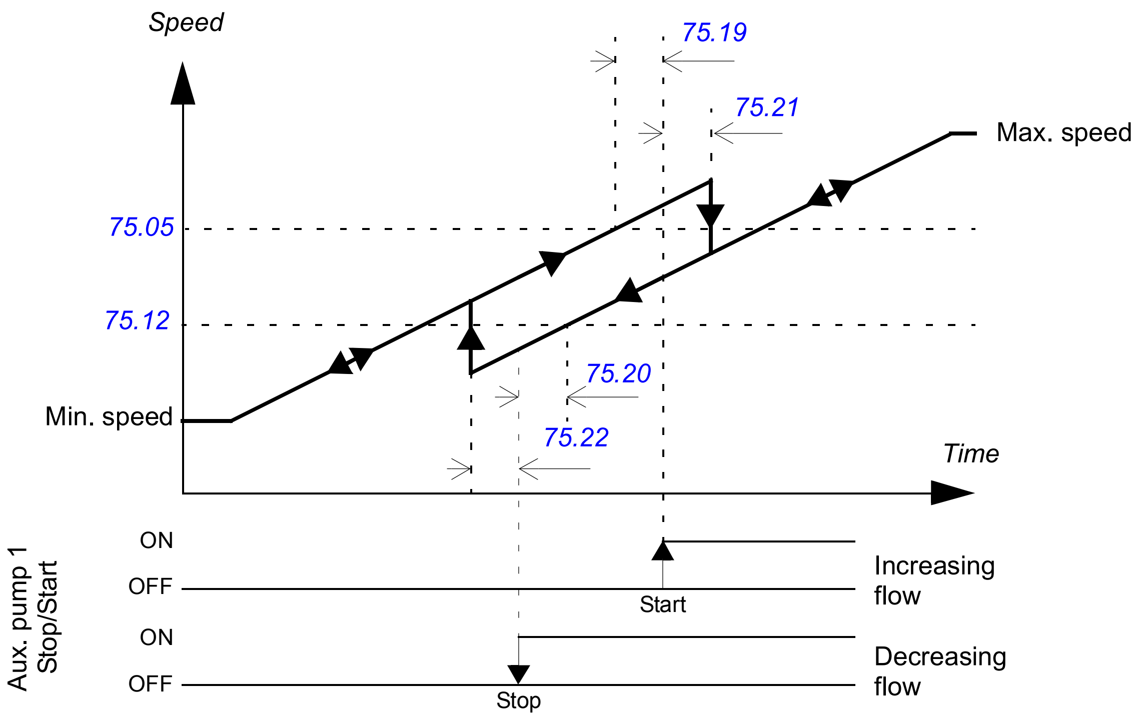

| | | | |
|---|---|---|---|
| | | In case the master status switches from one drive to another and the reference changes drastically, the drive compares the most recent reference value with the previous reference. If the difference between the references is more than 10%, the follower will accelerate/decelerate towards the new reference along a ramp. The acceleration and deceleration ramps are defined by parameters *75.26 Master speed acc* and *75.27 Master speed dec* respectively. The ramping will end when the new reference is reached. | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 75.04 | Follower ref | Only visible when *Multipump* is selected at parameter *75.01 Operation mode*.<br>Defines the reference used when parameter *75.03 Follower mode* is set to *Const speed*, and the drive is running as a follower.<br>The following diagram illustrates the starting of the drives in a typical multipump configuration as the reference (flow demand) first increases, then decreases. Follower start and stop delays (parameters *75.19 Start delay* and *75.20 Stop delay*) are ignored in this presentation. | |

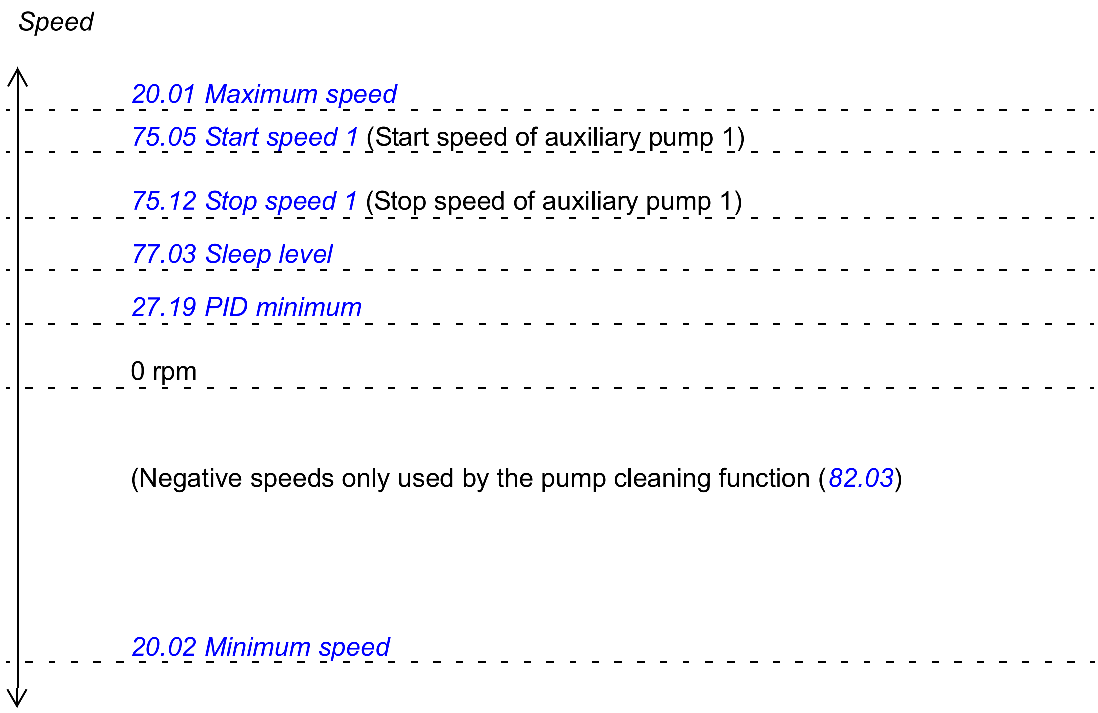



| | | | |
|---|---|---|---|
| | 0…32767 rpm | Reference setting. This should generally be set at the optimal operating point of the pump. | 1 = 1 rpm |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 75.05 | Start speed 1 | Defines the start speed for auxiliary pump 1.<br>When the speed of the pump connected directly to the drive exceeds this value and no auxiliary pumps are running, the start delay counter (see parameter *75.19 Start delay*) is started. If the speed is still at the same level or higher when the delay elapses, the first auxiliary pump starts.<br>The running speed of the drive is decreased by *Start speed 1* - *Stop speed 1* after the auxiliary pump starts. | |



| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | | The following diagram shows the order of some common speeds in a pump application. | |



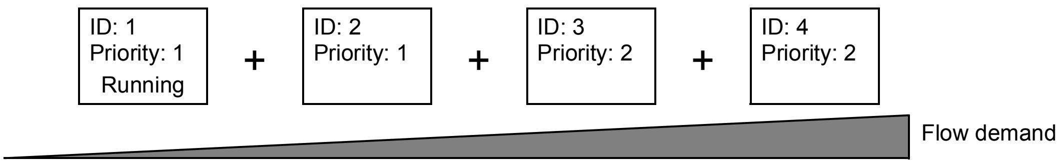

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 0…32767 rpm | Start speed for auxiliary pump 1. | 1 = 1 rpm |
| 75.06 | Start speed 2 | Defines the start speed for auxiliary pump 2. See parameter *75.05 Start speed 1*. | |
| | 0…32767 rpm | Start speed for auxiliary pump 2. | 1 = 1 rpm |
| 75.07 | Start speed 3 | Defines the start speed for auxiliary pump 3. See parameter *75.05 Start speed 1*. | |
| | 0…32767 rpm | Start speed for auxiliary pump 3. | 1 = 1 rpm |
| 75.08 | Start speed 4 | Defines the start speed for auxiliary pump 4. See parameter *75.05 Start speed 1*. | |
| | 0…32767 rpm | Start speed for auxiliary pump 4. | 1 = 1 rpm |
| 75.09 | Start speed 5 | Defines the start speed for auxiliary pump 5. See parameter *75.05 Start speed 1*. | |
| | 0…32767 rpm | Start speed for auxiliary pump 5. | 1 = 1 rpm |
| 75.10 | Start speed 6 | Defines the start speed for auxiliary pump 6. See parameter *75.05 Start speed 1*. | |
| | 0…32767 rpm | Start speed for auxiliary pump 6. | 1 = 1 rpm |
| 75.11 | Start speed 7 | Defines the start speed for auxiliary pump 7. See parameter *75.05 Start speed 1*. | |
| | 0…32767 rpm | Start speed for auxiliary pump 7. | 1 = 1 rpm |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 75.12 | Stop speed 1 | Defines the stop speed for auxiliary pump 1.<br>When the speed of the pump connected directly to the drive falls below this value and one auxiliary pump is running, the stop delay counter (see parameter *75.20 Stop delay*) is started. If the speed is still at the same level or lower when the delay elapses, the first auxiliary pump stops.<br>The running speed of the drive is increased by *Start speed 1* - *Stop speed 1* after the auxiliary pump stops.<br>See also parameter *75.05 Start speed 1*. | |
| | 0…32767 rpm | Stop speed for auxiliary pump 1. | 1 = 1 rpm |
| 75.13 | Stop speed 2 | Defines the stop speed for auxiliary pump 2. See parameter *75.12 Stop speed 1*. | |
| | 0…32767 rpm | Stop speed for auxiliary pump 2. | 1 = 1 rpm |
| 75.14 | Stop speed 3 | Defines the stop speed for auxiliary pump 3. See parameter *75.12 Stop speed 1*. | |
| | 0…32767 rpm | Stop speed for auxiliary pump 3. | 1 = 1 rpm |
| 75.15 | Stop speed 4 | Defines the stop speed for auxiliary pump 4. See parameter *75.12 Stop speed 1*. | |
| | 0…32767 rpm | Stop speed for auxiliary pump 4. | 1 = 1 rpm |
| 75.16 | Stop speed 5 | Defines the stop speed for auxiliary pump 5. See parameter *75.12 Stop speed 1*. | |
| | 0…32767 rpm | Stop speed for auxiliary pump 5. | 1 = 1 rpm |
| 75.17 | Stop speed 6 | Defines the stop speed for auxiliary pump 6. See parameter *75.12 Stop speed 1*. | |
| | 0…32767 rpm | Stop speed for auxiliary pump 6. | 1 = 1 rpm |
| 75.18 | Stop speed 7 | Defines the stop speed for auxiliary pump 7. See parameter *75.12 Stop speed 1*. | |
| | 0…32767 rpm | Stop speed for auxiliary pump 7. | 1 = 1 rpm |
| 75.19 | Start delay | Defines a start delay for auxiliary pumps. See parameter *75.05 Start speed 1*. | |
| | 0…12600 s | Start delay. | 1 = 1 s |
| 75.20 | Stop delay | Defines a stop delay for auxiliary pumps. See parameter *75.05 Start speed 1*. | |
| | 0…12600 s | Stop delay. | 1 = 1 s |
| 75.21 | Speed hold on | See diagram at parameter *75.05 Start speed 1*. | |
| | 0…100 s | Speed hold time for auxiliary pump switch-on. | 1 = 1 s |
| 75.22 | Speed hold off | See diagram at parameter *75.05 Start speed 1*. | |
| | 0…100 s | Speed hold time for auxiliary pump switch-off. | 1 = 1 s |
| 75.23 | Min pumps allow | Defines the minimum number of pumps that will run simultaneously.<br>**Note:** The pumps that are kept running will ignore the stop speeds defined for them by other parameters in this group. | |
| | 0…8 | Minimum number of pumps. | 1 = 1 |
| 75.24 | Max pumps allow | Defines the maximum number of pumps that can be run simultaneously. | |
| | 0…8 | Maximum number of pumps. | 1 = 1 |

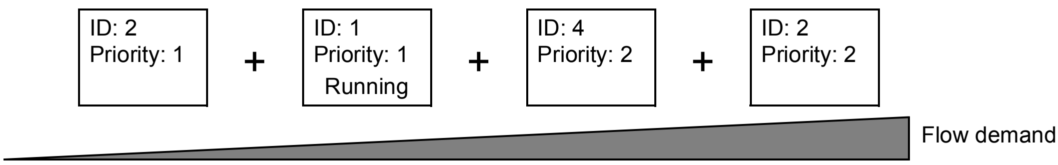
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 75.25 | Drive start dly | Start delay for the pump that is directly controlled by the drive. This does not affect the starting of the auxiliary pumps.<br>**WARNING!** There must always be a delay set if the pumps are equipped with star-delta starters. The delay must be set longer than the time setting of the starter. After the pump is switched on by the relay output of the drive, there must be enough time for the star-delta starter to first switch to star and then back to delta before the pump is connected to the drive. | |
| | 0…600 s | Start delay for drive-controlled pump. | 1 = 1 s |
| 75.26 | Master speed acc | Defines the acceleration time in case the latest reference received by the drive is higher than the previous reference. This is likely to happen when the master status is passed on from one drive to another. The parameter sets the ramp-up time as seconds from zero to maximum frequency (**not** from previous reference to new reference).<br>The parameter is effective only in the *Copy of mstr* and *Master speed* follower modes. See parameter *75.03 Follower mode*. | |
| | 0…1800 s | Acceleration time. | 1 = 1 s |
| 75.27 | Master speed dec | Defines the acceleration time in case the latest reference received by the drive is lower than the previous reference. This is likely to happen when the master status is passed on from one drive to another. The parameter sets the ramp-down time as seconds from maximum frequency to zero (**not** from previous reference to new reference).<br>The parameter is effective only in the *Copy of mstr* and *Master speed* follower modes. See parameter *75.03 Follower mode*. | |
| | 0…1800 s | Deceleration time. | 1 = 1 s |
| **76 MF communication** | | Communication configuration for applications consisting of multiple pumps with dedicated drives. | |
| 76.01 | Enable MF comm | Enables/disables drive-to-drive communication through the D2D link.<br>**Note:** Drive-to-drive communication can only be enabled if the embedded fieldbus interface is disabled (parameter *58.01 Protocol ena sel* is set to *Disabled*). | |
| | No | Drive-to-drive communication disabled. | 0 |
| | Yes | Drive-to-drive communication enabled. | 1 |
| 76.02 | Pump node | Node number of the drive on the drive-to-drive link.<br>**Notes:**<br>• Each drive on the link must have a unique node number.<br>• If the drive is not given a priority class, the node number is also used in determining the starting order of pumps. | |
| | 0…8 | Node number. | 1 = 1 |
| 76.03 | Master enable | Determines (or defines a source that determines) if the drive is allowed to be master on the drive-to-drive link. | |
| | No | The drive can only be a follower on the drive-to-drive link. | 0 |
| | Yes | The drive is allowed to be master on the drive-to-drive link. | 1 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 76.04 | Pump prior sel | Defines a source that chooses a start priority for the drive. Two preset priorities are available: either can be selected permanently, or a digital signal source used to switch between the two presets.<br>Please note that the Autochange feature will attempt to equalize the duty between drives with the same priority rather than between drives with different priorities.<br>With a digital source,<br>0 = priority defined by *76.05 Prior choice 1*<br>1 = priority defined by *76.06 Prior choice 2*. | |
| | Choice 1 | Start priority defined by parameter *76.05 Prior choice 1*. | 0 |
| | Choice 2 | Start priority defined by parameter *76.06 Prior choice 2*. | 1 |
| | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 76.05 | Prior choice 1 | Priority preset 1. See parameter *76.04 Pump prior sel*. | |
| | 1…4 | Priority preset 1. | 1 = 1 |
| 76.06 | Prior choice 2 | Priority preset 2. See parameter *76.04 Pump prior sel*. | |
| | 1…4 | Priority preset 2. | 1 = 1 |
| 76.07 | Mstr loss action | If the drive is a follower, cannot find a master on the drive-to-drive link, and is not itself allowed to be master, it will wait for the delay specified by parameter *76.08 Mstr loss delay*, then proceed as defined by this parameter. The drive will also generate an alarm. | |
| | Const speed | The drive continues running and adopts the speed defined by parameter *26.08 Const speed3*. | 0 |
| | Last speed | The drive continues running at the last valid reference received from the master. | 1 |
| 76.08 | Mstr loss delay | Delay for a master loss situation. See parameter *76.07 Mstr loss action*. | |
| | 0…3600 s | Delay for master loss. | 1 = 1 s |

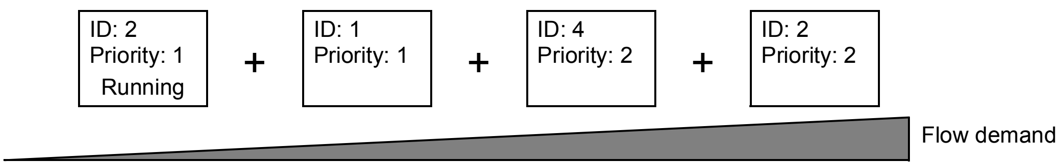
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 76.09 | Start order corr | Whenever the application requires more pumping volume, additional drives are started. The starting order is dependent on the priority setting of the drive (parameters *76.04*…*76.06*). Whenever several drives have the same priority, the one with the lowest node number (parameter *76.02*) is started first by default.<br>The Autochange function can be used to automatically rotate the starting order within each priority group. Drives running before the Autochange may continue to run so that the new starting order cannot be applied immediately; this parameter defines the method with which the drive order of priority is corrected.<br>*Example:*<br>One pump is running. If necessary, additional pumps are started in the following order: | |




While there is constant flow demand (and a pump must be running), the Autochange function is activated, rotating the starting order within each priority. After Autochange, the order is as follows:




The desired order, however, is this:



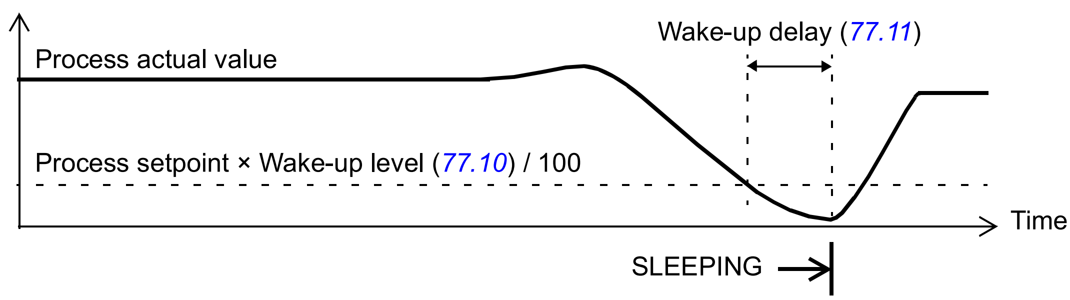

The selections of this parameter define how the desired order is achieved.

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Optimal | Drive order of priority is corrected only when the number of drives needs to be increased or decreased by the master as required by the process. | 0 |
| | Instant chng | Drive order of priority is corrected as soon as a new starting order is generated, for example when the Autochange conditions are met. The order is corrected by stopping low-priority drives. Higher-priority drives are then started as required by the process. | 1 |
| 76.10 | Master location | Defines whether the master status is passed on with each started drive or not. | |
| | Stable | The first drive started will remain the master as long as possible, until, for example, the drive is no longer allowed to be master (by parameter *76.03 Master enable*), or the drive trips on a fault. | 0 |

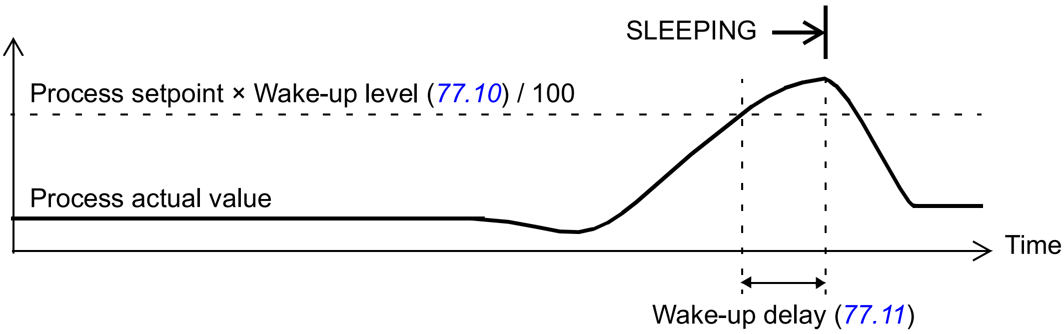
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | In start | The drive that was started last, and is allowed to be master by parameter *76.03 Master enable*, is the master. | 1 |
| 76.11 | Shared IO enable | Determines whether shared signals broadcast on the drive-to-drive link (if any) are received by the drive. | |
| | No | Shared signals not received. | 0 |
| | Yes | Shared signals received. The signals received are shown by parameters *02.43 Shared signal 1* and *02.44 Shared signal 2*. | 1 |
| 76.12 | Set as source | Determines whether the drive broadcasts shared signals on the drive-to-drive link or not. | |
| | No | The drive does not broadcast shared signals. | 0 |
| | Yes | The drive broadcasts the signals selected by parameters *76.13 Shared signal 1* and *76.14 Shared signal 2* as shared signals on the drive-to-drive link. | 1 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 76.13 | Shared signal 1 | Selects a signal to be broadcast as shared signal 1 on the drive-to-drive link. | |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | Proc act | *04.01 Act val* (see page *123*). | 1073742849 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 76.14 | Shared signal 2 | Selects a signal to be broadcast as shared signal 2 on the drive-to-drive link. | |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | Setpoint | *04.02 Setpoint* (see page *123*). | 1073742850 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 76.15 | Share lost actn | Defines the action taken by the drive if no shared signals are received for the time defined by parameter *76.16 Share lost delay*. (This parameter is only effective if parameter *76.11 Shared IO enable* is set to *Yes*.) | |
| | Alarm | The drive generates an alarm, *MF NO SHARED DATA*. | 0 |
| | Fault | The drive trips on a fault, *MF NO SHARED DATA*. | 1 |
| | Const speed | The drive continues running and adopts the speed defined by parameter *26.08 Const speed3*. | 2 |
| | Last speed | The drive continues running at the last valid reference received from the master. | 3 |
| 76.16 | Share lost delay | Delay for a shared signal loss situation. See parameter *76.15 Share lost actn*. | |
| | 0…3600 s | Delay for shared signal loss. | 1 = 1 s |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| **77 Pump sleep** | | Sleep function settings.<br>See also section *Sleep function* on page *59*. | |
| 77.01 | Sleep mode sel | Enables/disables the sleep function. | |
| | Not used | Sleep function disabled. | 0 |
| | Internal | The signal selected by parameter *77.02 Sleep int sel* is compared to the value of *77.03 Sleep level*. If the signal remains below this value longer than the sleep delay (*77.04 Sleep delay*), the drive shifts to sleep mode.<br>The sleep and wake-up delays (*77.04 Sleep delay* and *77.11 Wake up delay*) are in force. | 1 |
| | External | The sleep function is activated by the source selected by parameter *77.05 Sleep ext sel*.<br>The sleep delay (*77.04 Sleep delay*) is not in force but the wake-up delay (*77.11 Wake up delay*) is. | 2 |
| | Int+ext | When the source selected by parameter *77.05 Sleep ext sel* is “1”, the sleep function works as with the setting *Internal*. When the source selected by parameter *77.05 Sleep ext sel* is “0”, the sleep function is disabled. | 3 |
| | Soft ext | When the source selected by parameter *77.05 Sleep ext sel* is “0”, the sleep function is disabled.<br>When the source selected by parameter *77.05 Sleep ext sel* is “1”, the input of the PID controller is set to 0. After the drive enters sleep mode, it will not wake up until the signal returns to “0”. | 4 |
| 77.02 | Sleep int sel | Selects the internal signal to be monitored by the sleep function when parameter *77.01 Sleep mode sel* is set to *Internal*, *Int+ext* or *Soft ext*. | |
| | Speed | *01.01 Motor speed rpm* (see page *112*). | 1073742081 |
| | Speed % | *01.02 Motor speed %* (see page *112*). | 1073742082 |
| | AI1 | *02.04 AI1* (see page *113*). | 1073742340 |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 | *02.06 AI2* (see page *113*). | 1073742342 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | Proc act | *04.01 Act val* (see page *123*). | 1073742849 |
| | Flow act | *05.05 Flow act* (see page *125*). | 1073743109 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 77.03 | Sleep level | Defines the start limit for the sleep function when parameter *77.01 Sleep mode sel* is set to *Internal*, *Int+ext* or *Soft ext*. | |
| | -32768.00 … 32768.00 | Sleep start level. | 100 = 1 |
| 77.04 | Sleep delay | Defines the delay for the sleep start function. See parameter *77.03 Sleep level*. When the monitored signal falls below the sleep level, the counter starts. When the signal exceeds the sleep level, the counter resets. | |
| | 0 … 12600 s | Sleep start delay. | 1 = 1 s |

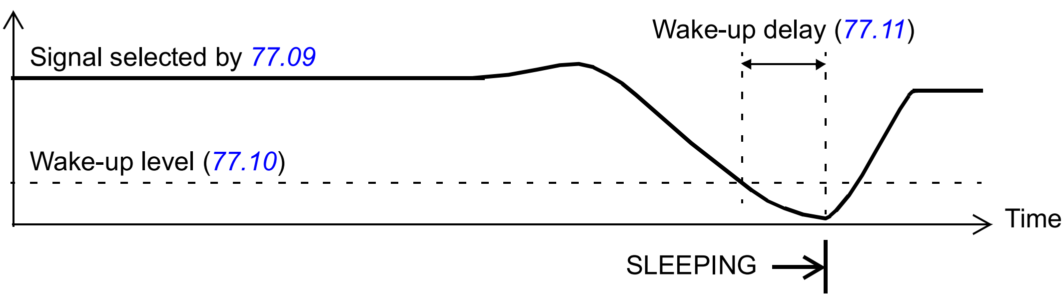
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 77.05 | Sleep ext sel | Defines a source that is used by parameter *77.01 Sleep mode sel*, selections *External*, *Int+ext* and *Soft ext*. See the descriptions of those selections for the usage of this signal source. | |
| | Not used | No source selected. | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 77.06 | Sleep boost step | When the drive is entering sleep mode, the setpoint is increased by this percentage for the time defined by parameter *77.07 Sleep boost time*.<br>No auxiliary pumps are started. If active, sleep boost is aborted when the drive wakes up. See the diagram in section *Sleep function* (starting page *59*). | |
| | 0.00 … 32767.00 % | Sleep boost step. | 100 = 1% |
| 77.07 | Sleep boost time | Sets the boost time for the sleep boost step defined by parameter *77.06 Sleep boost step*. | |
| | 0…100 s | Sleep boost time. | 1 = 1 s |
| 77.08 | Wake up mode sel | Selects the signal that is compared to the wake up level *77.10 Wake up level*, and the condition that must be true for the drive to wake up.<br>If the selected condition does not remain true until the wake-up delay (*77.11 Wake up delay*) expires, the delay counter is reset. | |
| | Wake > ref | If the process actual value (see group *28 Procact sel*) remains below the process setpoint (see group *29 Setpoint sel*) multiplied by the wake-up level for longer than the wake-up delay (*77.11 Wake up delay*), the drive wakes up. See the diagram below. | 0 |



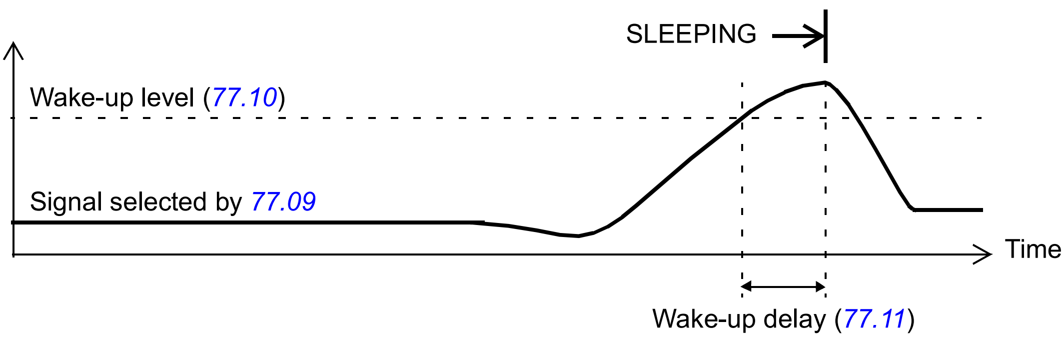
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Wake < ref | If the process actual value (see group *28 Procact sel*) remains above the process setpoint (see group *29 Setpoint sel*) multiplied by the wake-up level for longer than the wake-up delay (*77.11 Wake up delay*), the drive wakes up. See the diagram below. | 1 |




| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Wake > ext | If the signal selected by parameter *77.09 Wake up ext src* remains below the wake-up level (*77.10 Wake up level*) longer than the wake-up delay (*77.11 Wake up delay*), the drive wakes up. | 2 |




| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Wake < ext | If the signal selected by parameter *77.09 Wake up ext src* remains above the wake-up level (*77.10 Wake up level*) longer than the wake-up delay (*77.11 Wake up delay*), the drive wakes up. | 3 |




| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 77.09 | Wake up ext src | Selects the signal source for parameter *77.09 Wake up ext src*, selections *Wake > ext* and *Wake < ext*. | |
| | AI1 | *02.04 AI1* (see page *113*). | 1073742340 |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 | *02.06 AI2* (see page *113*). | 1073742342 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | Proc act | *04.01 Act val* (see page *123*). | 1073742849 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Flow act | *05.05 Flow act* (see page *125*). | 1073743109 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 77.10 | Wake up level | Defines the wake-up limit for the sleep function. See the selections of parameter *77.08 Wake up mode sel*. | |
| | -32768.00 … 32767.00 | Wake-up level. | 100 = 1 |
| 77.11 | Wake up delay | Defines the wake-up delay for the sleep function. See the selections of parameter *77.08 Wake up mode sel*. | |
| | 0 … 100 s | Wake-up delay. | 1 = 1 s |

| **78 Pump autochange** | | Pump Autochange and interlock settings.<br>See also section *Autochange* page *61*. | |
|---|---|---|---|
| 78.01 | Autochg style | Selects whether the Autochange function is used. | |
| | No | Autochange disabled. The drive with the lowest node number is started first. | 0 |
| | Fixed | Autochange will occur at intervals defined by parameter *78.05 Autochg interval* provided that the drive speed is below the value defined by parameter *78.04 Autochg level*.<br>**Note:** The timing is based on drive power-on time (rather than drive running time). | 1 |
| | Hourcount | The pumping duty is distributed among the pumps according to parameters *04.28 Pump runtime*, *78.14 Runtime change* and *78.15 Runtime diff*. | 2 |
| | All stop | Autochange will occur when all the pumps are stopped. | 3 |
| 78.02 | Autochg trad | Selects whether only auxiliary pumps or all pumps are affected by the Autochange function.<br>This parameter is only valid in traditional pump control. | |
| | All | All pumps are affected by the Autochange function. | 0 |
| | Aux | Only auxiliary (direct-on-line) pumps are affected by the Autochange function. | 1 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 78.03 | Interlock mode | Defines whether interlocks are used or not. This parameter is only valid in traditional pump control.<br>**WARNING!** Use of the Autochange function also requires the use of interlocks.<br>Interlocks are used in applications where one pump at a time is connected to the output of the drive. The remaining pumps are powered from the supply and started/stopped by the relay outputs of the drive.<br>A contact of the manual on/off switch (or protective device, such as a thermal relay) of each pump is wired to the selected interlock input. The logic will detect if the pump is unavailable and start the next available pump instead.<br>The interlock inputs are defined by parameters *78.06…78.13*.<br>If the interlock circuit of the speed-regulated pump (the pump connected to the drive output) is switched off, the pump is stopped and all relay outputs are de-energized. Then the drive will restart. The next available pump in the Autochange sequence will be started as the regulated pump.<br>If the interlock circuit of a direct-on-line pump is switched off, the drive will not try to start that pump until the interlock circuit is switched on again. The other pumps will operate normally. | |
| | Not used | Interlocks not used. | 0 |
| | On | Interlocks in use. | 1 |

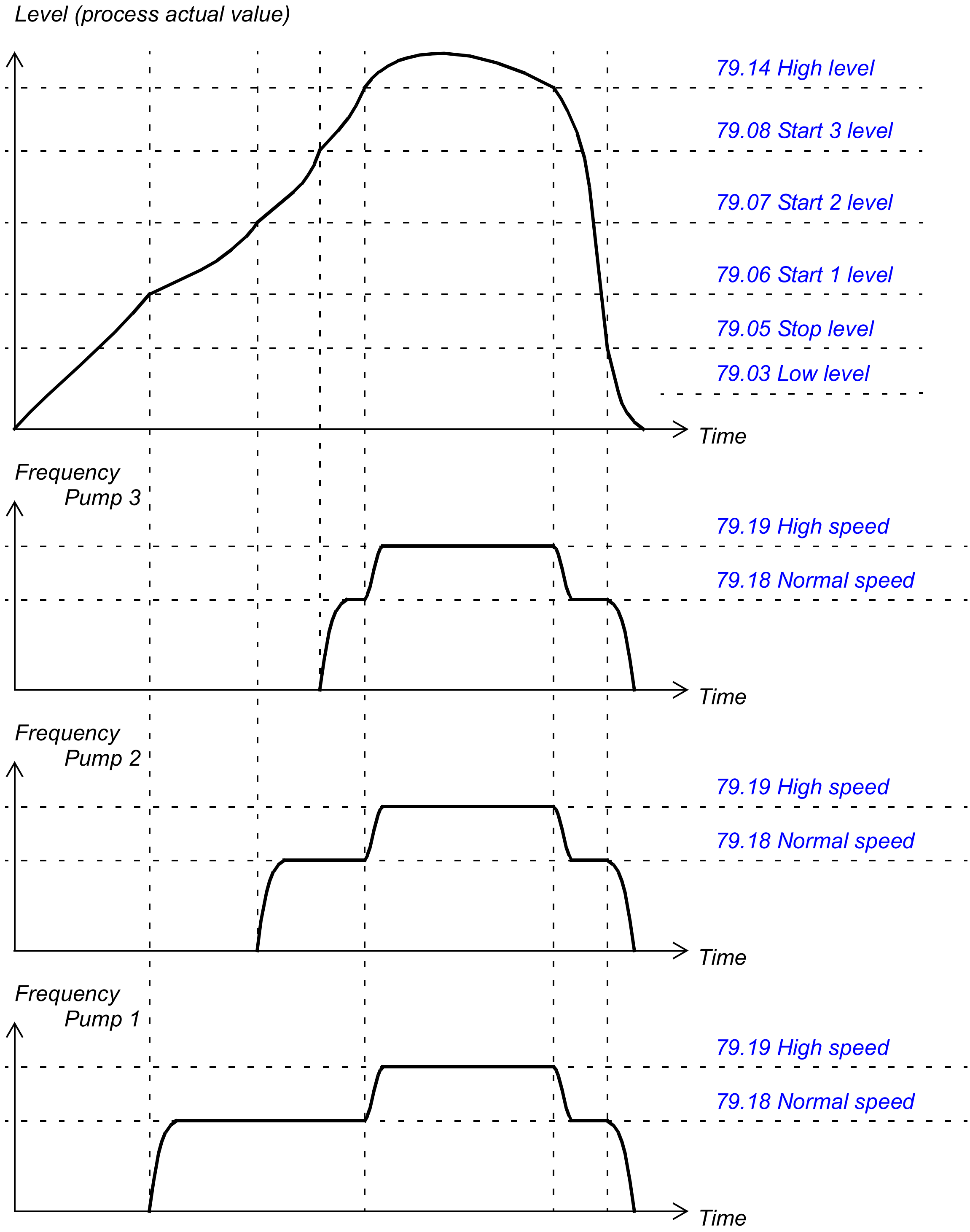
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 78.04 | Autochg level | Speed limit for the Autochange function when parameter *78.01 Autochg style* is set to *Fixed*.<br>This parameter is only valid in traditional pump control.<br>The pump starting sequence is changed when the Autochange interval has elapsed and the drive speed is below this limit. Autochanging is indicated by a warning on the control panel display.<br>**Notes:**<br>• The value of this parameter must be within the allowed range (between minimum and maximum limits). Otherwise no Autochanging is possible.<br>• When the drive is powered off, the values of the starting sequence counter and the Autochange interval counter are stored. The counters will continue from these values after the drive is powered on.<br>*Example:* There are three pumps in a system (parameter *75.02 Nbr of pumps* is set to 3). Autochange level is set to 1500 rpm.<br>An Autochange occurs when the drive speed is below 1500 rpm, and the Autochange interval has elapsed since the previous Autochange. Upon the Autochange,<br>1. All pumps are stopped<br>2. The starting sequence is incremented (from 1-2-3 to 2-3-1, etc.)<br>3. The contactor that controls the speed-regulated pump is closed<br>4. The delay set by parameter *75.25 Drive start dly* passes<br>5. The speed-regulated pump is energized and normal operation starts.<br>If the Autochange level is 0 rpm and the interval has elapsed, Autochange will occur during a stop (for example, when the Sleep function is active). | |
| | 0…32767 rpm | Autochange level. | 1 = 1 rpm |
| 78.05 | Autochg interval | Specifies the Autochange interval. See parameter *78.04 Autochg level*. | |
| | 0.00 … 1092.25 h | Autochange interval. | 100 = 1 h |
| 78.06 | Interlock pump 1 | Selects the input (or signal) for status of pump 1.<br>When the input is 1, the drive assumes the pump is in use and can be started. | |
| | Not used | The interlock is off, meaning that the pump is not in use. | 0 |
| | On | The interlock is on, meaning that the pump is ready for use. | 1 |
| | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 78.07 | Interlock pump 2 | Selects the input (or signal) for status of pump 2.<br>When the input is 1, the drive assumes the pump is in use and can be started. | |
| | Not used | The interlock is off, meaning that the pump is not in use. | 0 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | On | The interlock is on, meaning that the pump is ready for use. | 1 |
| | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 78.08 | Interlock pump 3 | Selects the input (or signal) for status of pump 3.<br>When the input is 1, the drive assumes the pump is in use and can be started. | |
| | Not used | The interlock is off, meaning that the pump is not in use. | 0 |
| | On | The interlock is on, meaning that the pump is ready for use. | 1 |
| | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 78.09 | Interlock pump 4 | Selects the input (or signal) for status of pump 4.<br>When the input is 1, the drive assumes the pump is in use and can be started. | |
| | Not used | The interlock is off, meaning that the pump is not in use. | 0 |
| | On | The interlock is on, meaning that the pump is ready for use. | 1 |
| | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 78.10 | Interlock pump 5 | Selects the input (or signal) for status of pump 5.<br>When the input is 1, the drive assumes the pump is in use and can be started. | |
| | Not used | The interlock is off, meaning that the pump is not in use. | 0 |
| | On | The interlock is on, meaning that the pump is ready for use. | 1 |
| | 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 |

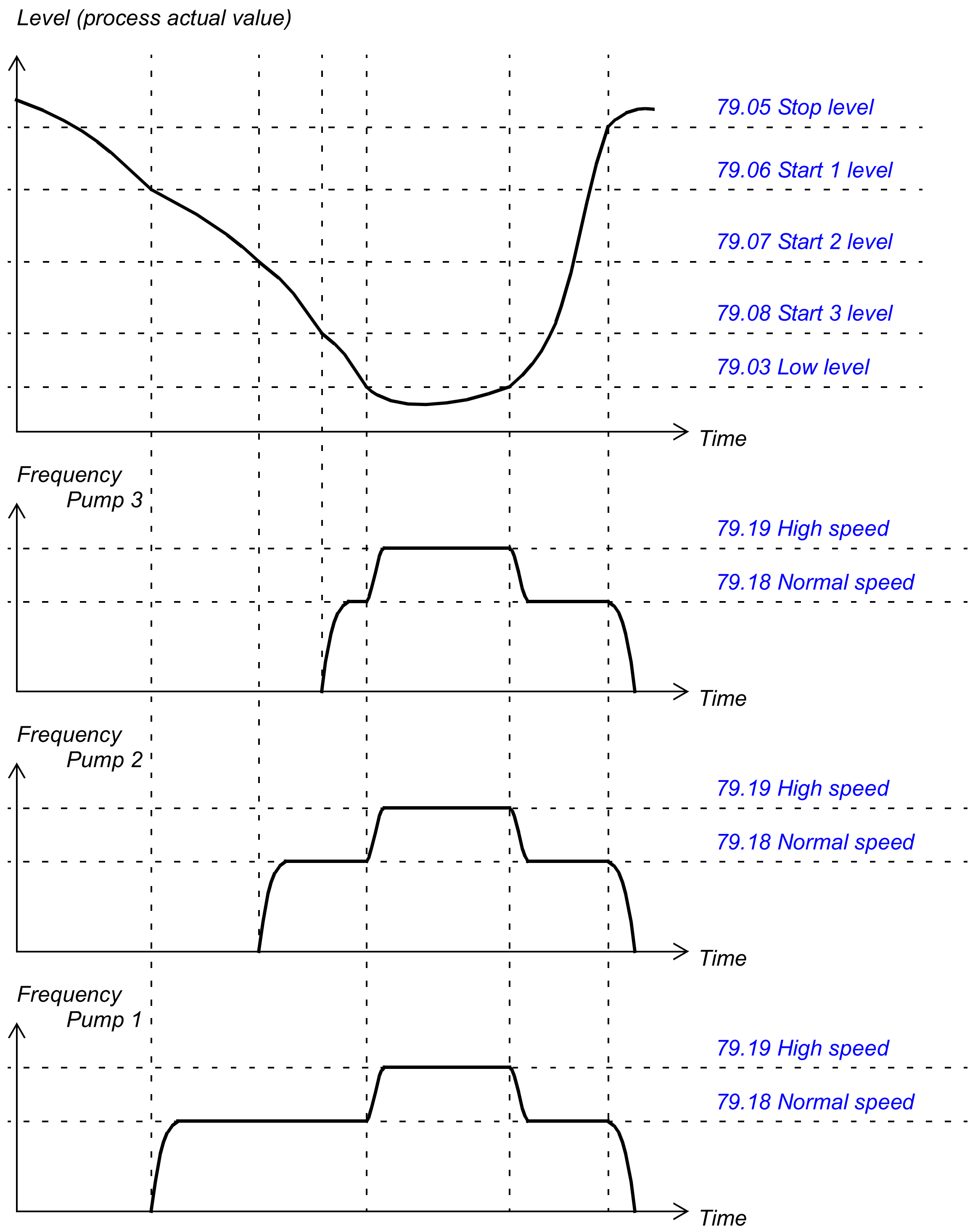
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 78.11 | Interlock pump 6 | Selects the input (or signal) for status of pump 6.<br>When the input is 1, the drive assumes the pump is in use and can be started. | |
| | Not used | The interlock is off, meaning that the pump is not in use. | 0 |
| | On | The interlock is on, meaning that the pump is ready for use. | 1 |
| | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 78.12 | Interlock pump 7 | Selects the input (or signal) for status of pump 7.<br>When the input is 1, the drive assumes the pump is in use and can be started. | |
| | Not used | The interlock is off, meaning that the pump is not in use. | 0 |
| | On | The interlock is on, meaning that the pump is ready for use. | 1 |
| | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 78.13 | Interlock pump 8 | Selects the input (or signal) for status of pump 8.<br>When the input is 1, the drive assumes the pump is in use and can be started. | |
| | Not used | The interlock is off, meaning that the pump is not in use. | 0 |
| | On | The interlock is on, meaning that the pump is ready for use. | 1 |
| | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 78.14 | Runtime change | Enables the reset, or arbitrary setting, of *04.28 Pump runtime*. | |
| | No | The parameter automatically reverts to this value. | 0 |
| | Set | Enables the setting of *04.28 Pump runtime* to an arbitrary value. | 1 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Reset | Resets parameter *04.28 Pump runtime*. | 2 |
| 78.15 | Runtime diff | Maximum pump runtime difference between drives. The control program will compare the value of the runtime counter (parameter *04.28 Pump runtime*) in each drive and attempt to keep the difference below this value. | |
| | 0 … 2147483647 h | Maximum runtime difference between drives. | 100 = 1 h |
| **79 Level control** | | Settings for level control applications.<br>See also section *Level control macro* (page *102*). | |
| 79.01 | Level mode | Defines whether the pump station is used for emptying or filling a container. | |
| | Off | Level control disabled. | 0 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Emptying | The pump station is used for emptying a container.<br>The diagram below shows the start, stop and supervision levels for emptying. For simplicity, only three pumps are shown. Parameter *79.02 Stopping mode* is assumed to be set to *Common stop*; *79.16 Start stop delay* is assumed to be set to 0.00 seconds. | 1 |



| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Filling | The pump station is used for filling a container.<br>The diagram below shows the start, stop and supervision levels for filling. For simplicity, only three pumps are shown. Parameter *79.02 Stopping mode* is assumed to be set to *Common stop*; *79.16 Start stop delay* is assumed to be set to 0.00 seconds. | 2 |

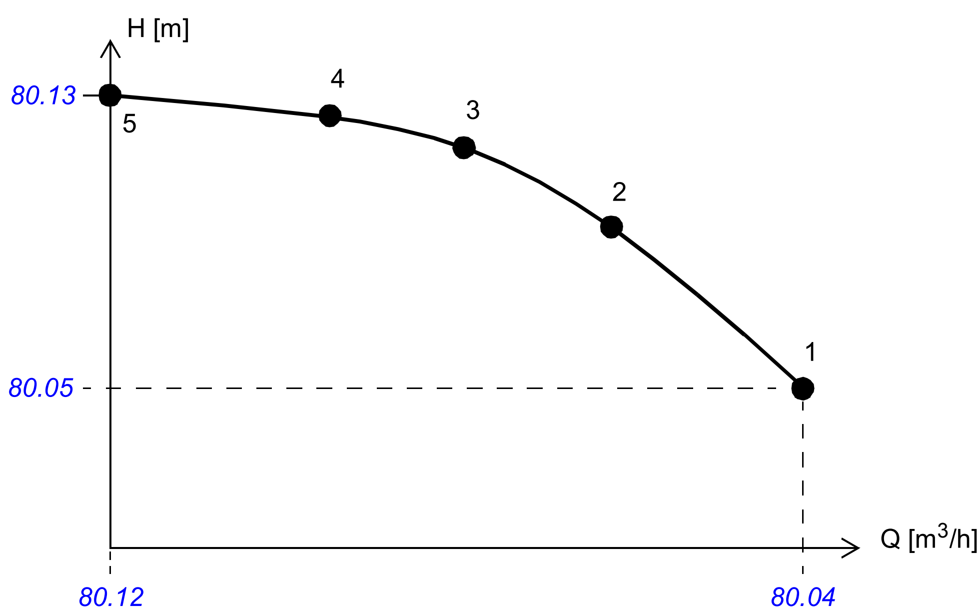


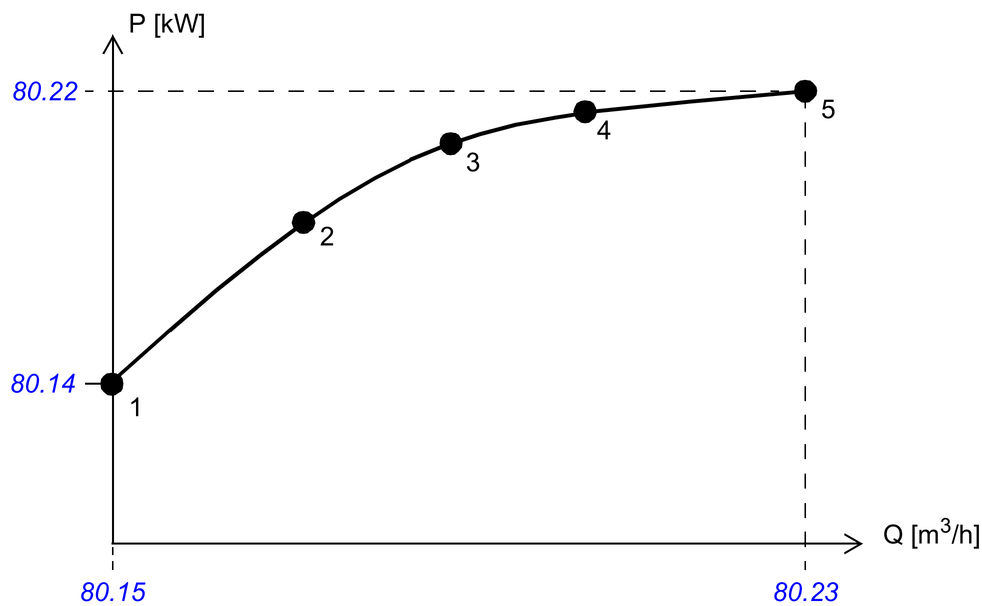
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 79.02 | Stopping mode | Selects whether the pumps are stopped simultaneously or individually. | |
| | Stable level | When the start level of a pump (parameters *79.06 Start 1 level*…*79.13 Start 8 level*) is reached, the master drive waits for the level delay (parameter *79.16 Start stop delay*) to elapse, then stops the pump. | 0 |
| | Common stop | All the pumps running will continue to run until the stop level (parameter *79.05 Stop level*) is reached. All pumps will then be stopped one by one at intervals defined by parameter *79.16 Start stop delay*. | 1 |
| 79.03 | Low level | Defines the low level for level control.<br>In emptying mode, when the measured level falls below the low level, all pumps stop (if not stopped already).<br>In filling mode, when the measured level falls below the low level, all pumps start running at the speed defined by parameter *79.19 High speed*.<br>See the diagrams at parameter *79.01 Level mode*. | |
| | 0.00 … 32768.00% | Low level. | TBA |
| 79.04 | Low switch | Selects a digital source that is used to determine that the liquid level in the container has fallen very low. When the source becomes active (1), an alarm, *LC TANK EMPTY* is given. The alarm is cleared when the source switches off. | |
| | Not used | No low switch used. | 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 |
| | Const<br>Pointer | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 79.05 | Stop level | Defines the stop level for the pump station.<br>If parameter *79.02 Stopping mode* is set to *Stable level*, pumps 3 and 2, for example, are stopped when *79.08 Start 3 level* and *79.07 Start 2 level* are reached respectively; pump 1 is stopped at the stop level.<br>If parameter *79.02 Stopping mode* is set to *Common stop*, all pumps will continue to run until the stop level is reached.<br>See the diagrams at parameter *79.01 Level mode*. | |
| | 0.00 … 32768.00% | Stop level. | TBA |
| 79.06 | Start 1 level | Defines the start level for pump 1. See the diagrams at parameter *79.01 Level mode*. | |
| | 0.00 … 32768.00% | Start level for pump 1. | TBA |
| 79.07 | Start 2 level | Defines the start level for pump 2. This is also the stop level for the pump unless *Common stop* is selected at parameter *79.02 Stopping mode*. See the diagrams at parameter *79.01 Level mode*. | |
| | 0.00 … 32768.00% | Start level for pump 2. | TBA |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 79.08 | Start 3 level | Defines the start level for pump 3. This is also the stop level for the pump unless *Common stop* is selected at parameter *79.02 Stopping mode*. See the diagrams at parameter *79.01 Level mode*. | |
| | 0.00 … 32768.00% | Start level for pump 3. | TBA |
| 79.09 | Start 4 level | Defines the start level for pump 4. This is also the stop level for the pump unless *Common stop* is selected at parameter *79.02 Stopping mode*. See the diagrams at parameter *79.01 Level mode*. | |
| | 0.00 … 32768.00% | Start level for pump 4. | TBA |
| 79.10 | Start 5 level | Defines the start level for pump 5. This is also the stop level for the pump unless *Common stop* is selected at parameter *79.02 Stopping mode*. See the diagrams at parameter *79.01 Level mode*. | |
| | 0.00 … 32768.00% | Start level for pump 5. | TBA |
| 79.11 | Start 6 level | Defines the start level for pump 6. This is also the stop level for the pump unless *Common stop* is selected at parameter *79.02 Stopping mode*. See the diagrams at parameter *79.01 Level mode*. | |
| | 0.00 … 32768.00% | Start level for pump 6. | TBA |
| 79.12 | Start 7 level | Defines the start level for pump 7. This is also the stop level for the pump unless *Common stop* is selected at parameter *79.02 Stopping mode*. See the diagrams at parameter *79.01 Level mode*. | |
| | 0.00 … 32768.00% | Start level for pump 7. | TBA |
| 79.13 | Start 8 level | Defines the start level for pump 8. This is also the stop level for the pump unless *Common stop* is selected at parameter *79.02 Stopping mode*. See the diagrams at parameter *79.01 Level mode*. | |
| | 0.00 … 32768.00% | Start level for pump 8. | TBA |
| 79.14 | High level | In emptying mode, when the measured level exceeds this value, all pumps start running at the speed defined by parameter *79.19 High speed*.<br>In filling mode, when the measured level exceeds this value, all pumps stop (if not stopped already). | |
| | 0.00 … 32768.00% | High level. | TBA |
| 79.15 | High switch | Selects a digital source that is used to determine that the liquid level in the container has risen very high. When the source switches on, an alarm, *LC TANK FULL* is given. The alarm is cleared when the source switches off. | |
| | Not used | No high switch used. | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 79.16 | Start stop delay | Sets a delay for stopping and starting a pump (or pumps). Whenever a start or stop level is reached, this delay must elapse before any action is taken. | |
| | 0 … 3600 s | Start/stop delay. | 1 = 1 s |
| 79.17 | Random coef | Randomizes the start levels (parameters *79.06*…*79.13*) to avoid caking on the walls of the container. For example, with this parameter set to 10.0%, the actual start level is randomized in the range of (start level - 10%) … (start level + 10%). | |
| | 0.0 … 10.0% | Random coefficient. | TBA |
| 79.18 | Normal speed | In emptying mode, defines the pump speed when the measured level is below the high level setting (parameter *79.14*), and the high switch (parameter *79.15*) is not active.<br>In filling mode, defines the pump speed when the measured level is above the low level setting (parameter *79.03*), and low switch (parameter *79.04*) is not active.<br>Ideally, this parameter should be set at the optimal operating point of the pump. | |
| | 0.0…32767.0 rpm | Normal running speed. | TBA |
| 79.19 | High speed | In emptying mode, defines the pump speed when the measured level exceeds the level defined by parameter *79.14 High level*, or when the high limit switch (parameter *79.15*) is active.<br>In filling mode, sets the pump speed when the measured level falls below the level defined by parameter *79.03 Low level*, or when the low limit switch (parameter *79.04*) is active.<br>See the diagrams at parameter *79.01 Level mode*. | |
| | 0.0…32767.0 rpm | High running speed. | TBA |

| | | | |
|---|---|---|---|
| **80 Flow calculation** | | Settings for the flow calculation function.<br>See also section *Flow calculation* on page *62*. | |
| 80.01 | Flow calc mode | Enables the flow calculation function, and determines whether a PQ (power/flow) curve or HQ (head/flow) curve is used for the calculation. The curves are defined by parameters *80.04*…*80.23*. | |
| | Not used | Flow calculation not used. | 0 |
| | PQ curve | The PQ curve is used for flow calculation. | 1 |
| | HQ curve | The HQ curve is used for flow calculation. | 2 |
| | Both | Both the HQ and PQ curves are used for flow calculation. The transition point between the curves is set by parameter *80.24 HQ PQ brk point*. | 3 |
| 80.02 | Pump inlet sel | Selects the analog input (or other signal source) used for pump inlet pressure measurement. | |
| | Zero | No input selected (no pressure sensor available). | 0 |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | AI3 scaled | *02.09 AI3 scaled* (see page *113*). | 1073742345 |
| | AI4 scaled | *02.11 AI4 scaled* (see page *114*). | 1073742347 |
| | AI5 scaled | *02.13 AI5 scaled* (see page *114*). | 1073742349 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 80.03 | Pump outlet sel | Selects the analog input (or other signal source) used for pump outlet pressure measurement. | |
| | Zero | No input selected (no pressure sensor available). | 0 |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | AI3 scaled | *02.09 AI3 scaled* (see page *113*). | 1073742345 |
| | AI4 scaled | *02.11 AI4 scaled* (see page *114*). | 1073742347 |
| | AI5 scaled | *02.13 AI5 scaled* (see page *114*). | 1073742349 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 80.04 | HQ curve Q1 | Flow rate (in cubic meters per hour) at point 1 of the HQ performance curve.<br>Parameters *80.04*…*80.13* define the HQ performance curve of the pump for the flow calculation function. The H (head, or level) and Q (flow rate) coordinates of five points on the curve are entered. The values are provided by the pump manufacturer. All points defined should lie within the practical operating range of the pump.<br>Below is an example of an HQ performance curve. The defining parameters of the first and last points are shown.<br><br> | |
| | 0.00 … 32767.00 $m^3/h$ | Flow rate at point 1 of the HQ curve. | 100 = 1 $m^3/h$ |
| 80.05 | HQ curve H1 | Head (in meters) at point 1 of the HQ performance curve. | |
| | 0.00 … 32767.00 m | Head at point 1 of the HQ curve. | 100 = 1 m |
| 80.06 | HQ curve Q2 | Flow rate (in cubic meters per hour) at point 2 of the HQ performance curve. | |
| | 0.00 … 32767.00 $m^3/h$ | Flow rate at point 2 of the HQ curve. | 100 = 1 $m^3/h$ |

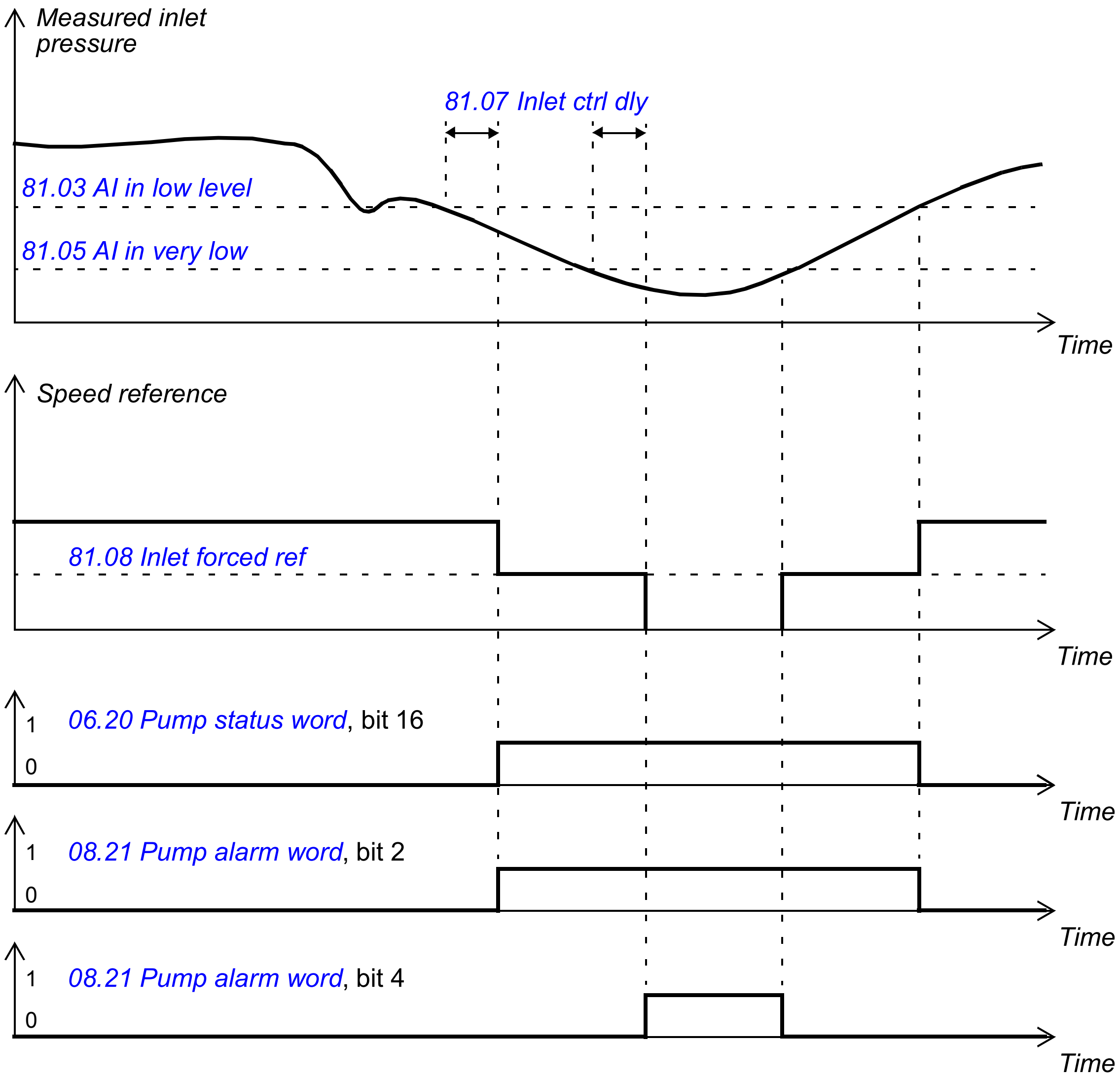
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 80.07 | HQ curve H2 | Head (in meters) at point 2 of the HQ performance curve. | |
| | 0.00 … 32767.00 m | Head at point 2 of the HQ curve. | 100 = 1 m |
| 80.08 | HQ curve Q3 | Flow rate (in cubic meters per hour) at point 3 of the HQ performance curve. | |
| | 0.00 … 32767.00 $m^3/h$ | Flow rate at point 3 of the HQ curve. | 100 = 1 $m^3/h$ |
| 80.09 | HQ curve H3 | Head (in meters) at point 3 of the HQ performance curve. | |
| | 0.00 … 32767.00 m | Head at point 3 of the HQ curve. | 100 = 1 m |
| 80.10 | HQ curve Q4 | Flow rate (in cubic meters per hour) at point 4 of the HQ performance curve. | |
| | 0.00 … 32767.00 $m^3/h$ | Flow rate at point 4 of the HQ curve. | 100 = 1 $m^3/h$ |
| 80.11 | HQ curve H4 | Head (in meters) at point 4 of the HQ performance curve. | |
| | 0.00 … 32767.00 m | Head at point 4 of the HQ curve. | 100 = 1 m |
| 80.12 | HQ curve Q5 | Flow rate (in cubic meters per hour) at point 5 of the HQ performance curve. | |
| | 0.00 … 32767.00 $m^3/h$ | Flow rate at point 5 of the HQ curve. | 100 = 1 $m^3/h$ |
| 80.13 | HQ curve H5 | Head (in meters) at point 5 of the HQ performance curve. | |
| | 0.00 … 32767.00 m | Head at point 5 of the HQ curve. | 100 = 1 m |
| 80.14 | PQ curve P1 | Power input (in kilowatts) of pump at point 1 on the PQ performance curve.<br>Parameters *80.14*…*80.23* define the PQ performance curve of the pump for the flow calculation function. The P (power input) and Q (flow rate) coordinates of five points on the curve are entered. The values are provided by the pump manufacturer. All points defined should lie within the practical operating range of the pump.<br>Below is an example of an PQ performance curve. The defining parameters of the first and last points are shown.<br> | |
| | 0.00 … 32767.00 kW | Power input of pump at point 1. | 100 = 1 kW |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 80.15 | PQ curve Q1 | Flow rate (in cubic meters per hour) at point 1 on the PQ performance curve. | |
| | 0.00 … 32767.00 $m^3/h$ | Flow rate at point 1 of the PQ curve. | 100 = 1 $m^3/h$ |
| 80.16 | PQ curve P2 | Power input (in kilowatts) of pump at point 2 on the PQ performance curve. | |
| | 0.00 … 32767.00 kW | Power input of pump at point 2. | 100 = 1 kW |
| 80.17 | PQ curve Q2 | Flow rate (in cubic meters per hour) at point 2 on the PQ performance curve. | |
| | 0.00 … 32767.00 $m^3/h$ | Flow rate at point 2 of the PQ curve. | 100 = 1 $m^3/h$ |
| 80.18 | PQ curve P3 | Power input (in kilowatts) of pump at point 3 on the PQ performance curve. | |
| | 0.00 … 32767.00 kW | Power input of pump at point 3. | 100 = 1 kW |
| 80.19 | PQ curve Q3 | Flow rate (in cubic meters per hour) at point 3 on the PQ performance curve. | |
| | 0.00 … 32767.00 $m^3/h$ | Flow rate at point 3 of the PQ curve. | 100 = 1 $m^3/h$ |
| 80.20 | PQ curve P4 | Power input (in kilowatts) of pump at point 4 on the PQ performance curve. | |
| | 0.00 … 32767.00 kW | Power input of pump at point 4. | 100 = 1 kW |
| 80.21 | PQ curve Q4 | Flow rate (in cubic meters per hour) at point 4 on the PQ performance curve. | |
| | 0.00 … 32767.00 $m^3/h$ | Flow rate at point 4 of the PQ curve. | 100 = 1 $m^3/h$ |
| 80.22 | PQ curve P5 | Power input (in kilowatts) of pump at point 5 on the PQ performance curve. | |
| | 0.00 … 32767.00 kW | Power input of pump at point 5. | 100 = 1 kW |
| 80.23 | PQ curve Q5 | Flow rate (in cubic meters per hour) at point 5 on the PQ performance curve. | |
| | 0.00 … 32767.00 $m^3/h$ | Flow rate at point 5 of the PQ curve. | 100 = 1 $m^3/h$ |
| 80.24 | HQ PQ brk point | Sets the transition point between the HQ and PQ performance curves. The PQ curve is used above this breakpoint. | |
| | 0.00 … 32767.00 m | Head breakpoint between HQ and PQ curves. | 100 = 1 m |
| 80.25 | Pump inlet diam | The diameter of the pump inlet in meters. | |
| | 0.00 … 32767.00 m | Pump inlet diameter. | 100 = 1 m |
| 80.26 | Pump outlet diam | The diameter of the pump outlet in meters. | |
| | 0.00 … 32767.00 m | Pump outlet diameter. | 100 = 1 m |
| 80.27 | Sensors hgt diff | Defines the height difference between the inlet and outlet pressure sensors. | |
| | 0.00 … 32767.00 m | Height difference. | 100 = 1 m |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 80.28 | Pump nom speed | Defines the nominal speed of the pump in rpm. | |
| | 0…32767 rpm | Nominal speed of pump. | 1 = 1 rpm |
| 80.29 | Density | Defines the density of the fluid to be pumped for the flow calculation function. | |
| | 0.00 … 32767.00 $kg/m^3$ | Fluid density. | 100 = 1 $kg/m^3$ |
| 80.30 | Efficiency | Total efficiency of the motor/pump combination. | |
| | 0.00 … 100.00% | Efficiency. | 100 = 1% |
| 80.31 | Flow calc gain | Flow calculation gain for possible calculation correction. | |
| | 0.00 … 32767.00 | Calculation correction gain. | 100 = 1 |
| 80.32 | Calc low sp | Defines a speed limit below which flow is not calculated. | |
| | 0…32767 rpm | Low speed limit for flow calculation. | 1 = 1 rpm |
| 80.33 | Sum flow reset | Resets the total calculated flow counter (parameter *05.08*). | |
| | No | No reset. | 0 |
| | Reset | Reset the counter. | 1 |

| **81 Pump protection** | | Settings for pump protection functions.<br>See also section *Protective functions* on page *64*. | |
|---|---|---|---|
| 81.01 | Inlet prot ctrl | Enables the primary supervision of pump inlet pressure and selects the action taken when low inlet pressure is detected. The selected action is taken only after the measured pressure has remained below the pressure limit (*81.03 AI in low level*) for longer than the value of parameter *81.07 Inlet ctrl dly*.<br>The pressure can be measured using an analog pressure sensor or a pressure switch.<br>The input for an analog sensor is defined by parameter *81.02 AI measure inlet*. With an analog sensor, a separate action for "very low inlet pressure" can be defined using parameter *81.05 AI in very low*.<br>The input for a pressure switch is defined by parameter *81.06 DI status inlet*. | |
| | Not used | Primary inlet pressure supervision not used. | 0 |
| | Alarm | Detection of low inlet pressure produces an alarm after the delay defined by parameter *81.07 Inlet ctrl dly* expires. | 1 |
| | Fault | Detection of low inlet pressure trips the drive after the delay defined by parameter *81.07 Inlet ctrl dly* expires. | 2 |

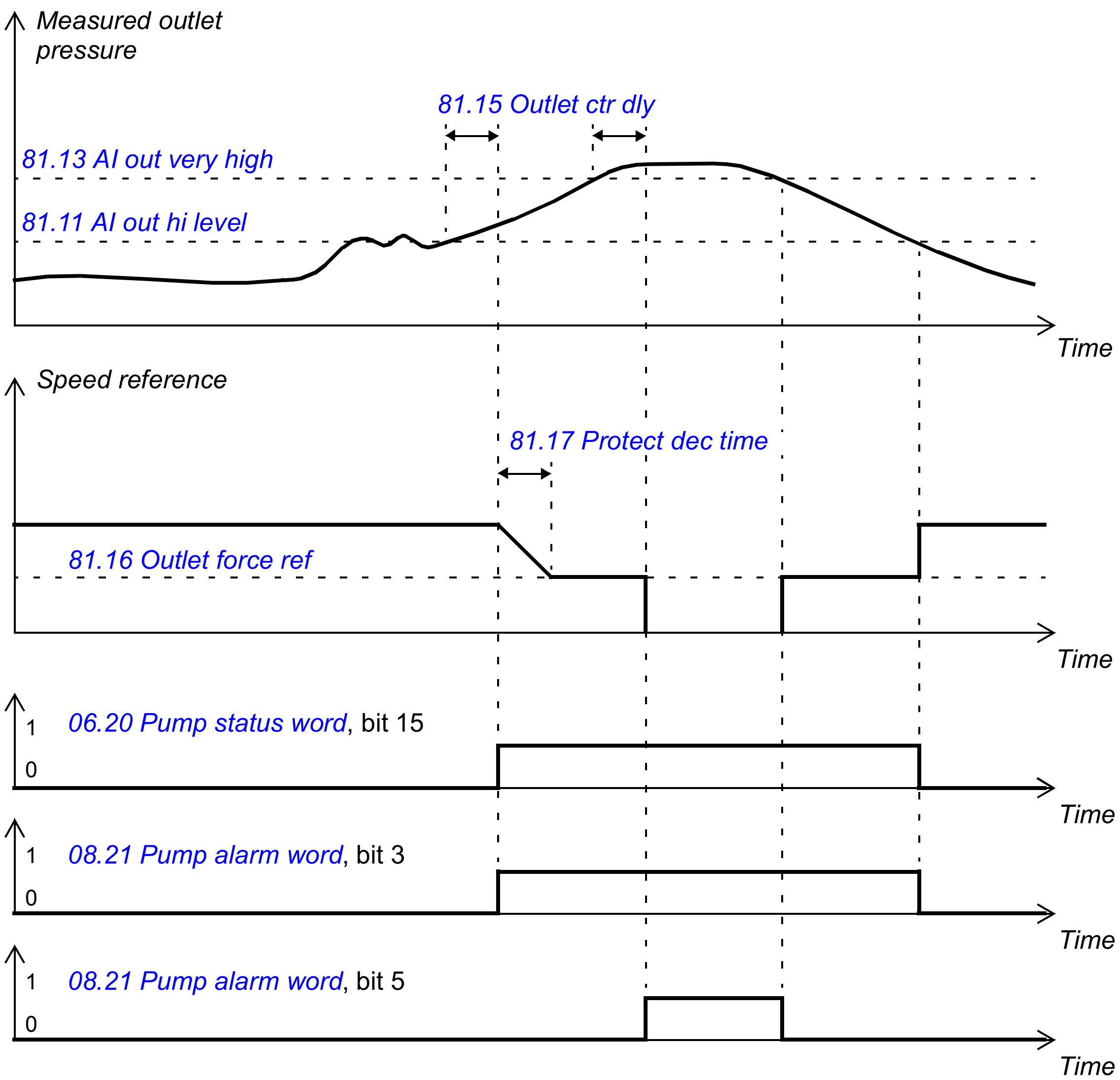
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Protect | Detection of low inlet pressure produces an alarm after the delay defined by parameter *81.07 Inlet ctrl dly* expires. The pump speed is reduced to the speed defined by *81.08 Inlet forced ref*. | 3 |




| | | | |
|---|---|---|---|
| 81.02 | AI measure inlet | Selects the analog input (or signal source) for pump inlet pressure measurement. | |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | AI3 scaled | *02.09 AI3 scaled* (see page *113*). | 1073742345 |
| | AI4 scaled | *02.11 AI4 scaled* (see page *114*). | 1073742347 |
| | AI5 scaled | *02.13 AI5 scaled* (see page *114*). | 1073742349 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 81.03 | AI in low level | Pressure limit for primary inlet pressure supervision. See parameter *81.01 Inlet prot ctrl*. | |
| | 0.00 … 32767.00 bar | Pressure limit. | 100 = 1 bar |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 81.04 | Very low ctrl | Enables the secondary supervision of pump inlet pressure, and selects the action taken after very low inlet pressure is detected. The selected action is taken only after the measured pressure has remained below the pressure limit (*81.05 AI in very low*) for longer than the value of parameter *81.07 Inlet ctrl dly*. See the diagram at parameter *81.01 Inlet prot ctrl*.<br>**Note:** With a pressure switch, this parameter has no effect. | |
| | Not sel | Secondary inlet pressure supervision not used. | 0 |
| | Fault | Detection of very low inlet pressure trips the drive. | 1 |
| | Stop | Detection of very low inlet pressure stops the drive. The drive will restart if the pressure rises above the limit. | 2 |
| 81.05 | AI in very low | Pressure limit for secondary inlet pressure supervision. See parameter *81.04 Very low ctrl*. | |
| | 0.00 … 32767.00 bar | Pressure limit. | 100 = 1 bar |
| 81.06 | DI status inlet | Selects the digital input for connection of a pressure switch at the pump inlet. The “normal” state is 1. If the selected input switches to 0, the action defined by parameter *81.01 Inlet prot ctrl* is taken after the delay set by parameter *81.07 Inlet ctrl dly* expires. | |
| | Not used | No pressure switch connected. | 1 |
| | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 81.07 | Inlet ctrl dly | Delay for primary and secondary supervision of pump inlet pressure. See parameter *81.01 Inlet prot ctrl*. | |
| | 0…600 s | Delay. | 1 = 1 s |
| 81.08 | Inlet forced ref | Pump speed reference for parameter *81.01 Inlet prot ctrl*, selection *Protect*. | |
| | 0.0 … 32767.0 rpm | Speed reference. | 10 = 1 rpm |
| 81.09 | Outlet prot ctrl | Enables the primary supervision of pump outlet pressure and selects the action taken when high outlet pressure is detected. The selected action is taken only after the measured pressure has remained above the pressure limit (*81.11 AI out hi level*) for longer than the value of parameter *81.15 Outlet ctr dly*.<br>The pressure can be measured using an analog pressure sensor or a pressure switch.<br>The input for an analog sensor is defined by parameter *81.10 AI meas outlet*. With an analog sensor, a separate action for “very high outlet pressure” can be defined using parameter *81.13 AI out very high*.<br>The input for a pressure switch is defined by parameter *81.14 DI status outlet*. | |
| | Not used | Primary outlet pressure supervision not used. | 0 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Alarm | Detection of high outlet pressure produces an alarm after the delay defined by parameter *81.15 Outlet ctr dly* expires. | 1 |
| | Fault | Detection of high outlet pressure trips the drive after the delay defined by parameter *81.15 Outlet ctr dly* expires. | 2 |
| | Protect | Detection of high outlet pressure produces an alarm after the delay defined by parameter *81.15 Outlet ctr dly* expires. The pump speed is reduced to the speed defined by *81.16 Outlet force ref* within the time defined by parameter *81.17 Protect dec time*. | 3 |




| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 81.10 | AI meas outlet | Selects the analog input (or signal source) for pump outlet pressure measurement. | |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |
| | AI3 scaled | *02.09 AI3 scaled* (see page *113*). | 1073742345 |
| | AI4 scaled | *02.11 AI4 scaled* (see page *114*). | 1073742347 |
| | AI5 scaled | *02.13 AI5 scaled* (see page *114*). | 1073742349 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |

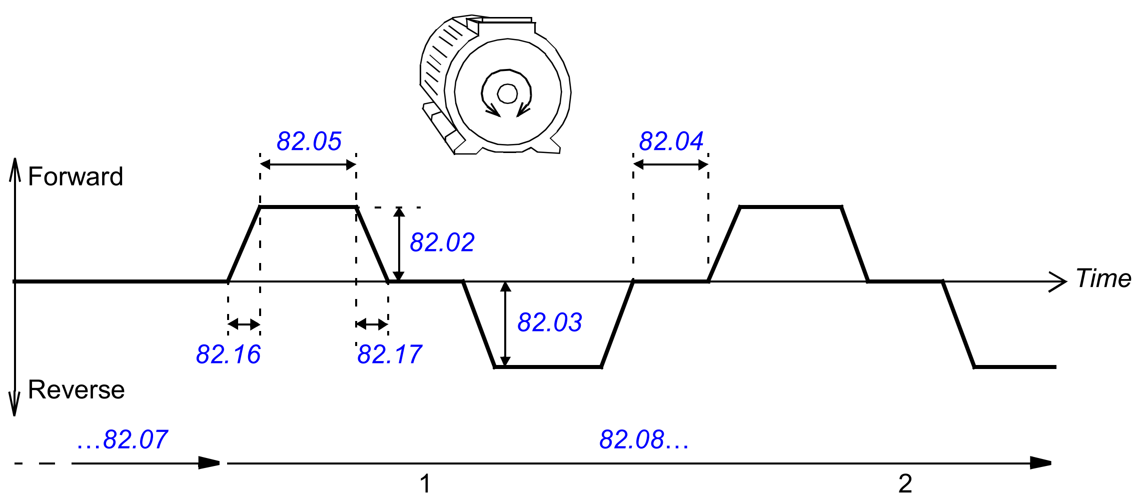
| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 81.11 | AI out hi level | Pressure limit for primary outlet pressure supervision. See parameter *81.09 Outlet prot ctrl*. | |
| | 0.00 … 32767.00 bar | Pressure limit. | 100 = 1 bar |
| 81.12 | Very high ctrl | Enables the secondary supervision of pump outlet pressure and selects the action taken when very high outlet pressure is detected. The selected action is taken only after the measured pressure has remained above the pressure limit (*81.13 AI out very high*) for longer than the value of parameter *81.15 Outlet ctr dly*. See the diagram at parameter *81.09 Outlet prot ctrl*.<br>**Note:** With a pressure switch, this parameter has no effect. | |
| | Not sel | Secondary outlet pressure supervision not used. | 0 |
| | Fault | Detection of very high outlet pressure trips the drive. | 1 |
| | Stop | Detection of very high outlet pressure stops the drive. The drive will restart if the pressure falls below the limit. | 2 |
| 81.13 | AI out very high | Pressure limit for secondary outlet pressure supervision. See parameter *81.12 Very high ctrl*. | |
| | 0.00 … 32767.00 bar | Pressure limit. | 100 = 1 bar |
| 81.14 | DI status outlet | Selects the digital input for connection of a pressure switch at the pump outlet. The “normal” state is 1. If the selected input switches to 0, the action defined by parameter *81.09 Outlet prot ctrl* is taken after the delay set by parameter *81.15 Outlet ctr dly* expires. | |
| | Not used | No pressure switch connected. | 1074070017 |
| | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 81.15 | Outlet ctr dly | Delay for primary and secondary supervision of pump outlet pressure. See parameter *81.09 Outlet prot ctrl*. | |
| | 0…600 s | Delay. | 1 = 1 s |
| 81.16 | Outlet force ref | Pump speed reference for parameter *81.09 Outlet prot ctrl*, selection *Protect*. | |
| | 0.0 … 32767.0 rpm | Speed reference. | 10 = 1 rpm |
| 81.17 | Protect dec time | PID controller ramp-down time for parameter *81.09 Outlet prot ctrl*, selection *Protect*. | |
| | 0…18000 s | PID controller ramp-down time for outlet pressure supervision. | 1 = 1 s |
| 81.18 | Flow source sel | Selects an source for flow measurement for minimum/maximum flow protection. See parameters *81.19 Flow max prot* and *81.21 Flow min prot*. | |
| | AI1 scaled | *02.05 AI1 scaled* (see page *113*). | 1073742341 |
| | AI2 scaled | *02.07 AI2 scaled* (see page *113*). | 1073742343 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | AI3 scaled | *02.09 AI3 scaled* (see page *113*). | 1073742345 |
| | AI4 scaled | *02.11 AI4 scaled* (see page *114*). | 1073742347 |
| | AI5 scaled | *02.13 AI5 scaled* (see page *114*). | 1073742349 |
| | Flow act | Calculated flow as indicated by *05.05 Flow act* (see page *125*). | 1073743109 |
| | Pointer | Value pointer setting (see *Terms and abbreviations* on page *109*). | - |
| 81.19 | Flow max prot | Defines the action to be taken if the flow (signal selected by parameter *81.18 Flow source sel*) remains above the limit set by parameter *81.20 Flow max level* for longer than the time set by parameter *81.23 Flow ctrl delay*. | |
| | Not sel | Maximum flow protection disabled. | 0 |
| | Alarm | The drive generates alarm *MAX FLOW*. | 1 |
| | Fault | The drive trips on fault *MAX FLOW*. | 2 |
| 81.20 | Flow max level | Defines the maximum flow limit. See parameter *81.19 Flow max prot*. | |
| | 0.00 … 32767.00 $m^3/s$ | Maximum flow. | 100 = 1 $m^3/s$ |
| 81.21 | Flow min prot | Defines the action to be taken if the flow (signal selected by parameter *81.18 Flow source sel*) remains below the limit set by parameter *81.22 Flow min level* for longer than the time set by parameter *81.23 Flow ctrl delay*.<br>See also parameter *81.24 Flow check delay*. | |
| | Not sel | Minimum flow protection disabled. | 0 |
| | Alarm | The drive generates alarm *MIN FLOW*. | 1 |
| | Fault | The drive trips on fault *MIN FLOW*. | 2 |
| 81.22 | Flow min level | Defines the minimum flow limit. See parameter *81.21 Flow min prot*. | |
| | 0.00 … 32767.00 $m^3/s$ | Minimum flow. | 100 = 1 $m^3/s$ |
| 81.23 | Flow ctrl delay | Specifies a delay for minimum/maximum flow protection. See parameters *81.19 Flow max prot* and *81.21 Flow min prot*. | |
| | 0…12600 s | Delay for minimum/maximum flow protection. | 1 = 1 s |
| 81.24 | Flow check delay | After starting the drive, defines a period during which the minimum flow protection is disabled so that normal flow can be reached. | |
| | 0…12600 s | Start delay for minimum flow protection. | 1 = 1 s |
| 81.25 | Appl prot ctrl | Enables/disables the Application profile protection function, based on long-term monitoring of an internal signal. If the selected signal exceeds (and remains above) the supervision limit longer than the delay set by parameter *81.27 Prof limit dly*, the alarm *PROFILE HIGH* is generated and *08.21 Pump alarm word* bit 6 set to 1. | |
| | Not used | Application profile protection disabled. | 0 |
| | PID error | Signal *04.04 Process PID err* compared to value of parameter *81.26 Prof limit*. | 1 |
| | PID out | Signal *04.05 Process PID out* compared to value of parameter *81.26 Prof limit*. | 2 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 81.26 | Prof limit | Supervision limit for the Application profile protection. | |
| | 0.00 … 32767.00 % | Supervision limit. | 100 = 1% |
| 81.27 | Prof limit dly | Delay for the Application profile protection. | |
| | 0.00 … 35791394.11 h | Delay. | 100 = 1 h |
| 81.28 | Pipefill enable | Enables/disables (or selects a signal source that enables/disables) the Pipefill function when the drive is started.<br>1 = Enable Pipefill function. If the signal is removed before Pipefill is completed, Pipefill is aborted and normal PID control enabled. | |
| | Not used | Pipefill function disabled. | 0 |
| | Active | Pipefill function enabled. | 1 |
| | DI1 | The status of digital input DI1 (as indicated by *02.01 DI status*, bit 0) determines whether the Pipefill function is enabled or disabled. | 1073742337 |
| | DI2 | The status of digital input DI2 (as indicated by *02.01 DI status*, bit 1) determines whether the Pipefill function is enabled or disabled. | 1073807873 |
| | DI3 | The status of digital input DI3 (as indicated by *02.01 DI status*, bit 2) determines whether the Pipefill function is enabled or disabled. | 1073873409 |
| | DI4 | The status of digital input DI4 (as indicated by *02.01 DI status*, bit 3) determines whether the Pipefill function is enabled or disabled. | 1073938945 |
| | DI5 | The status of digital input DI5 (as indicated by *02.01 DI status*, bit 4) determines whether the Pipefill function is enabled or disabled. | 1074004481 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 81.29 | Pipefill step | Defines the speed step used for the Pipefill function, as well as the pump speed reference immediately after the Pipefill function is activated.<br>The speed step is added to the reference after the time defined by parameter *81.31 Act change delay* has elapsed and the change in process actual value defined by parameter *81.30 Req act change* has not been reached.<br>The PID controller reference ramp time is specified by parameter *27.32 Pipefill ref acc*. | |
| | 0…32767 rpm | Speed step for the Pipefill function. | 1 = 1 rpm |
| 81.30 | Req act change | Defines the requested change in process actual value within the time set by parameter *81.31 Act change delay*. | |
| | 0.00 … 100.00% | Requested change. | 100 = 1% |
| 81.31 | Act change delay | Defines the time that is waited after the process actual value is compared to the previous actual value.<br>If parameter *81.30 Req act change* is measured in the actual value, the speed reference stays as it is. If *81.30 Req act change* is not seen in the actual value, the value of parameter *81.29 Pipefill step* is added to the speed reference. | |
| | 0…100 s | Delay for actual value change. | 1 = 1 s |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 81.32 | Pid enable dev | Defines the process actual value level at which the Pipefill function is disabled and normal PID control is enabled. After the level is reached, the time defined by parameter *81.33 Pid enb dev dly* is allowed to pass before normal PID control is enabled. PID reference ramps are then observed (if set).<br>The value is given in percent of the maximum process actual value. | |
| | 0.00 … 100.00% | Pipefill / PID control breakpoint. | 100 = 1% |
| 81.33 | Pid enb dev dly | Delay for enabling PID control. See parameter *81.32 Pid enable dev*. | |
| | 0…12600 s | PID enable delay. | 1 = 1 s |
| 81.34 | Pipefill timeout | Defines the maximum allowed time for the Pipefill function. If the target process actual value (parameter *81.32 Pid enable dev*) is not reached within this time, the action defined by parameter *81.35 Pipefill flt ctr* is taken. | |
| | 0…12600 s | Maximum Pipefill time. | 1 = 1 s |
| 81.35 | Pipefill flt ctr | Defines the action for the Pipefill timeout (parameter *81.34 Pipefill timeout*). | |
| | Alarm | The drive generates alarm *PIPEFILL TIMEOUT*. | 0 |
| | Fault | The drive trips on fault *PIPEFILL TOUT*. | 1 |
| | Activate PID | Normal PID control is enabled. | 2 |

| | | | |
|---|---|---|---|
| **82 Pump cleaning** | | Settings for the pump cleaning sequence.<br>See also section *Pump cleaning* on page *63*. | |
| 82.01 | Pump clean trig | Enables the pump cleaning sequence for the drive, and defines the triggering conditions.<br>**WARNING!** Before enabling the pump cleaning sequence, ensure it can be performed safely with the connected equipment.<br>**Notes:**<br>• Multiple triggering conditions can exist simultaneously.<br>• The cleaning sequence observes the speed limits defined by parameters *20.01 Maximum speed* and *20.02 Minimum speed*.<br>• The drive must be started and its run enable signal present before the cleaning sequence can be started. | |



<table>
<tr><th>No.</th><th>Name/Value</th><th>Description</th><th>FbEq</th></tr>
<tr><td colspan="4">
<table>
<tr><th>Bit</th><th>Name</th><th>Function</th></tr>
<tr><td>0</td><td>Enabled</td><td>0 = No: Cleaning sequence disabled.<br>1 = Yes: Cleaning sequence enabled.</td></tr>
<tr><td>1</td><td>Master enb</td><td>0 = No: Cleaning sequence not allowed when the drive is master.<br>1 = Yes: Cleaning sequence allowed when the drive is master.</td></tr>
<tr><td>2</td><td>Follower enb</td><td>0 = No: Cleaning sequence not allowed when the drive is a follower.<br>1 = Yes: Cleaning sequence allowed when the drive is a follower.</td></tr>
<tr><td>3</td><td>Time trig</td><td>1 = Enable: Cleaning sequence starts periodically at intervals defined by parameter 82.07 Time trig.</td></tr>
<tr><td>4</td><td>Supervision</td><td>1 = Enable: The cleaning sequence is started whenever the signal selected by parameter 82.09 Supervis source exceeds the value of parameter 82.10 Supervis limit.</td></tr>
<tr><td>5</td><td>At start</td><td>1 = Enable: Cleaning sequence performed on every start command.</td></tr>
<tr><td>6</td><td>Trig ptr</td><td>1 = Enable: The cleaning sequence is performed when the signal selected by parameter 82.12 Trig pointer changes to 1.</td></tr>
</table>
</td></tr>
<tr><td>82.02</td><td>Fwd step</td><td>Forward step frequency for the cleaning sequence in percent of the value of parameter 19.01 Speed scaling.</td><td></td></tr>
<tr><td></td><td>0.0 … 100.0%</td><td>Forward step frequency.</td><td>10 = 1%</td></tr>
<tr><td>82.03</td><td>Rev step</td><td>Reverse step frequency for the cleaning sequence in percent of the value of parameter 19.01 Speed scaling.</td><td></td></tr>
<tr><td></td><td>0.0 … 100.0%</td><td>Reverse step frequency.</td><td>10 = 1%</td></tr>
<tr><td>82.04</td><td>Off time</td><td>Interval between forward and reverse steps during the cleaning sequence.</td><td></td></tr>
<tr><td></td><td>0…1000 s</td><td>Interval between steps.</td><td>1 = 1 s</td></tr>
<tr><td>82.05</td><td>Fwd step time</td><td>Duration of each forward step during the cleaning sequence.</td><td></td></tr>
<tr><td></td><td>0…1000 s</td><td>Duration of each forward step.</td><td>1 = 1 s</td></tr>
<tr><td>82.06</td><td>Rev step time</td><td>Duration of each reverse step during the cleaning sequence.</td><td></td></tr>
<tr><td></td><td>0…1000 s</td><td>Duration of each reverse step.</td><td>1 = 1 s</td></tr>
<tr><td>82.07</td><td>Time trig</td><td>Time between periodical cleaning sequences. See parameter 82.01 Pump clean trig, bit 3.</td><td></td></tr>
<tr><td></td><td>0.00 … 35791394.11 h</td><td>Time between cleaning sequences.</td><td>1 = 1 min</td></tr>
<tr><td>82.08</td><td>Nbr of steps</td><td>Number of forward-reverse step combinations to be performed during the cleaning sequence.</td><td></td></tr>
<tr><td></td><td>0…2147483647</td><td>Number of steps.</td><td>1 = 1</td></tr>
<tr><td>82.09</td><td>Supervis source</td><td>Defines a signal that triggers the cleaning sequence when it remains above the limit defined by parameter 82.10 Supervis limit for longer than the time set by parameter 82.11 Supervis delay.<br>See also parameter 82.01 Pump clean trig, bit 4.</td><td></td></tr>
<tr><td></td><td>Current A</td><td>01.04 Motor current (see page 112).</td><td>1073742084</td></tr>
<tr><td></td><td>Current %</td><td>01.05 Motor current % (see page 112).</td><td>1073742085</td></tr>
<tr><td></td><td>Pointer</td><td>Value pointer setting (see Terms and abbreviations on page 109).</td><td>-</td></tr>
<tr><td>82.10</td><td>Supervis limit</td><td>Defines a limit for the signal selected by parameter 82.09 Supervis source.</td><td></td></tr>
<tr><td></td><td>0.0 … 32767.0</td><td>Cleaning sequence triggering limit.</td><td>10 = 1</td></tr>
</table>

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 82.11 | Supervis delay | Delay for signal triggering of the cleaning sequence. See parameter *82.09 Supervis source*. | |
| | 0…600 s | Cleaning sequence triggering delay. | 1 = 1 s |
| 82.12 | Trig pointer | Cleaning sequence trigger input selection. See parameter *82.01 Pump clean trig*, bit 6. | |
| | Not used | No input selected. | 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 |
| | Const | Bit pointer setting (see *Terms and abbreviations* on page *109*). | - |
| | Pointer | | |
| 82.13 | Clean max ctrl | Defines the action taken if the maximum number of cleaning sequences (*82.14 Clean max number*) is exceeded within the time set by parameter *82.15 Clean max period*. Forced cleanings are disregarded. | |
| | Not sel | No action taken. | 0 |
| | Alarm | The drive generates an alarm, *MAX CLEANINGS*. | 1 |
| | Fault | The drive trips on fault *MAX CLEANINGS*. | 2 |
| 82.14 | Clean max number | Defines the maximum number of cleaning sequences within the time set by parameter *82.15 Clean max period*. | |
| | 0…30 | Maximum number of cleaning sequences. | 1 = 1 |
| 82.15 | Clean max period | Defines the time (ending now) within which cleaning sequences are counted. | |
| | 0.00 … 35791394.11 h | Time within which cleaning sequences are counted. | 100 = 1 h |
| 82.16 | Clean step acc | Defines the acceleration time from 0 rpm to the step frequency (parameters *82.02 Fwd step* and *82.03 Rev step*). | |
| | 0…32767 s | Step acceleration time. | 1 = 1 s |
| 82.17 | Clean step dec | Defines the deceleration time from the step frequency (parameters *82.02 Fwd step* and *82.03 Rev step*) to 0 rpm. | |
| | 0…32767 s | Step deceleration time. | 1 = 1 s |

| | | | |
|---|---|---|---|
| **83 Energy monitoring** | | Energy consumption monitoring settings. See also section *Energy consumption monitoring* on page *81*. | |
| 83.01 | Energy mon mode | Enables/disables, and selects the mode of, consumed energy monitoring. | |
| | Not used | Energy monitoring not in use. | 0 |
| | Limits | The current energy monitoring period is compared to the consumption limit set by parameter *83.03 kWh limit*. | 1 |
| | Previous | The current energy monitoring period (parameter *05.20 kWh current read*) is compared to the previous period (*05.21 kWh prev read*). | 2 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Average | The current energy monitoring period (parameter *05.20 kWh current read*) is compared to the average of the two previous periods (*05.21 kWh prev read* and *05.22 kWh posprev read*). | 3 |
| 83.02 | Mon period | Defines the length of an energy monitoring period. The first period starts when the drive is powered up. | |
| | 0.00 … 35791394.11 h | Length of monitoring period. | 1 = 1 min |
| 83.03 | kWh limit | Consumed energy limit for parameter *83.01 Energy mon mode*, selection *Limits*. | |
| | 0…2147483647 kWh | Energy limit. | 1 = 1 kWh |
| 83.04 | Mon tolerance | Tolerance for energy limit. The energy consumption may exceed the reference energy by this tolerance value until the action defined by parameter *83.05 Energy mon ctrl* is taken. | |
| | 0…2147483647 kWh | Tolerance. | 1 = 1 kWh |
| 83.05 | Energy mon ctrl | Defines the action that is taken if the energy consumption exceeds the tolerance limits. | |
| | Not sel | No action taken. | 0 |
| | Alarm | The drive generates alarm *ENERGY LIMIT*. | 1 |
| 83.06 | Energy reset | Resets the energy monitoring counters. | |
| | No | No reset. The parameter automatically reverts to this value after a reset. | 0 |
| | Period | Resets the periodic energy counters (parameters *05.20*…*05.22*). | 1 |
| | Month | Resets the monthly energy counters (parameters *05.23*…*05.35*). | 2 |

| | | | |
|---|---|---|---|
| **94 Ext IO conf** | | I/O extension configuration. | |
| 94.01 | Ext IO1 sel | Activates an I/O extension installed into Slot 1. Depending on the module used, enables<br>• digital input DI7<br>• digital input/outputs DIO3…DIO6<br>• analog inputs AI3…AI5<br>• analog output AO3 or<br>• relay outputs RO3…RO6. | |
| | None | No extension installed into Slot 1. | 0 |
| | FIO-01 | FIO-01 extension installed into Slot 1. Additional 4 × DIO and 2 × RO are in use. | 1 |
| | FIO-11 | FIO-11 extension installed into Slot 1. Additional 2 × DIO, 3 × AI and 1 × AO are in use. | 2 |
| | FIO-21 | FIO-21 extension installed into Slot 1. Additional 1 × DI, 1 × AI and 2 × RO are in use. | 3 |
| | FIO-31 | FIO-31 extension installed into Slot 1. Additional 4 × RO are in use. | 4 |

| | | | |
|---|---|---|---|
| **95 Hw configuration** | | Diverse hardware-related settings. | |
| 95.01 | Ctrl boardSupply | Selects how the drive control unit is powered. | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 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 the estimated value, an alarm *(COOLALARM)* or fault *(COOLING)* is generated. | |
| | 0…55 °C | Drive ambient temperature. | 1 = 1 °C |

| | | | |
|---|---|---|---|
| **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*…*97.12*.<br>Notes:<br>Parameter value is automatically set to zero when ID run is selected by parameter *99.13 IDrun mode*. The values of parameters *97.02*…*97.12* are updated according to the motor characteristics identified during the ID run.<br>This parameter cannot be changed while the drive is running. | |
| | NoUserPars | Parameters *97.02*…*97.12* inactive. | 0 |
| | UserMotPars | The values of parameters *97.02*…*97.12* are used in the motor model. | 1 |
| | UserPosOffs | Reserved. | 2 |
| | AllUserPars | Reserved. | 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. | |
| | 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. | |
| | 0.00000 … 10.00000 p.u. | Main inductance in per unit. | 100000 = 1 p.u. |
| 97.05 | SigmaL user | Defines the leakage inductance $\sigma L_S$. | |
| | 0.00000 … 1.00000 p.u. | Leakage inductance in per unit. | 100000 = 1 p.u. |
| 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. | |
| | 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. | |
| | 0.00 …100000.00 mH | Main inductance. | 100 = 1 mH |
| 97.12 | SigL user SI | Defines the leakage inductance $\sigma L_S$. | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 0.00 …100000.00 mH | Leakage inductance. | 100 = 1 mH |

| **99 Start-up data** | | Language selection, motor configuration and ID run settings. | |
|---|---|---|---|
| 99.01 | Language | Selects the language of the control panel displays.<br>**Note:** Not all languages listed below are necessarily supported. | |
| | English | English. | 0809 |
| | Deutsch | German. | 0407 |
| | Italiano | Italian. | 0410 |
| | Espanol | Spanish. | 040A |
| | Nederlands | Dutch. | 0413 |
| | Francais | French. | 040C |
| | Dansk | Danish. | 0406 |
| | Russki | Russian. | 0419 |
| | Polski | Polish. | 0415 |
| | Turkce | Turkish. | 041F |
| | Magyar | Hungarian. | 040E |
| 99.05 | Motor ctrl mode | Selects the motor control mode. | |
| | DTC | Direct torque control. This mode is suitable for most applications.<br>**Note:** Instead of direct torque control, use scalar control<br>• 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),<br>• if the nominal current of the motor is less than 1/6 of the nominal output current of the drive,<br>• if the drive is used with no motor connected (for example, for test purposes),<br>• 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.<br>**Note:** Correct motor run requires that the magnetizing current of the motor does not exceed 90% of the nominal current of the inverter.<br>See also section *Scalar motor control* on page *70*. | 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.<br>**Notes:**<br>• Correct motor run requires that the magnetizing current of the motor does not exceed 90% of the nominal current of the drive.<br>• This parameter cannot be changed while the drive is running. | |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | 0.0 … 6400.0 A | Nominal current of the motor. The allowable range is 1/6…2 × $I_{max}$ of the drive (0…2 × $I_{max}$ with scalar control mode). | 10 = 1 A |
| 99.07 | Mot nom voltage | 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.<br>**Notes:**<br>• 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.<br>• This parameter cannot be changed while the drive is running. | |
| | 1/6 … 2 × $U_N$ | Nominal voltage of the motor. | 10 = 1 V |
| 99.08 | Mot nom freq | Defines the nominal motor frequency.<br>**Note:** This parameter cannot be changed while the drive is running. | |
| | 5.0 … 500.0 Hz | Nominal frequency of the motor. | 10 = 1 Hz |
| 99.09 | Mot nom speed | Defines the nominal motor speed. The setting must match the value on the rating plate of the motor.<br>**Notes:**<br>• For safety reasons, after ID run, the maximum and minimum speed limits (parameters *16.17* and *16.17*) are automatically set to 1.2 times the value of this parameter.<br>• This parameter cannot be changed while the drive is running. | |
| | 0 … 30000 rpm | Nominal speed of the motor. | 1 = 1 rpm |
| 99.10 | Mot nom power | 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.<br>The unit is selected by parameter *16.17 Power unit*.<br>**Note:** This parameter cannot be changed while the drive is running. | |
| | 0.00 … 10000.00 kW | Nominal power of the motor. | 100 = 1 kW |
| 99.11 | Mot nom cosfii | Defines the cosphi of the motor for a more accurate motor model. Not obligatory; if set, should match the value on the rating plate of the motor.<br>**Note:** This parameter cannot be changed while the drive is running. | |
| | 0.00 … 1.00 | Cosphi of the motor. | 100 = 1 |
| 99.12 | Mot nom torque | Defines the nominal motor shaft torque for a more accurate motor model. Not obligatory.<br>**Note:** This parameter cannot be changed while the drive is running. | |
| | 0 … 2147483.647 Nm | Nominal motor torque. | 1000 = 1 N•m |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| 99.13 | IDrun mode | 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.<br>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 *Standstill*. In this case, the ID run must be performed.<br>**Notes:**<br>• ID run can only be performed in local control (i.e. when drive is controlled via PC tool or control panel).<br>• ID run cannot be performed if parameter *99.05 Motor ctrl mode* is set to *Scalar*.<br>• ID run must be performed every time any of the motor parameters (*99.06*…*99.12*) have been changed. Parameter is automatically set to *Standstill* after the motor parameters have been set.<br>• Ensure that possible Safe torque off and emergency stop circuits are closed during ID run.<br>• 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 | 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.<br>**Notes:**<br>• 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.<br>• Check the direction of rotation of the motor before starting the ID run. During the run, the motor will rotate in the forward direction.<br>**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! | 1 |

| No. | Name/Value | Description | FbEq |
|---|---|---|---|
| | Reduced | Reduced ID Run. This mode should be selected instead of the Normal ID Run if<br>• mechanical losses are higher than 20% (i.e. the motor cannot be de-coupled from the driven equipment), or if<br>• 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).<br>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).<br>**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.<br>**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! | 2 |
| | Standstill | Standstill ID run. The motor is injected with DC current. The motor shaft will not rotate.<br>**Note:** This mode should be selected only if the *Normal* or *Reduced* ID run is not possible due to the restrictions caused by the connected mechanics (e.g. with lift or crane applications). | 3 |
| | Autophasing | Reserved. | 4 |
| | Cur meas cal | Current offset and gain measurement calibration. The calibration will be performed at next start. | 5 |

7

# Additional parameter data

## What this chapter contains

This chapter lists the parameters with some additional data. For parameter descriptions, see chapter *Parameters* on page *109*.

## Terms and abbreviations

| Term | Definition |
|---|---|
| Actual signal | Signal measured or calculated by the drive. Can usually only be monitored but not adjusted; some counters can however be reset by entering a 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 | $\underbrace{\text{16-bit value}}_{\text{= integer value}}\ \underbrace{\text{16-bit value}}_{\text{= fractional value}}$ (31 bits + sign) |
| REAL24 | $\underbrace{\text{8-bit value}}_{\text{= integer value}}\ \underbrace{\text{24-bit value}}_{\text{= fractional value}}$ (31 bits + sign) |

| | |
|---|---|
| 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 addresses

Refer to the *User's Manual* of the fieldbus adapter.

# Pointer parameter format in fieldbus communication

Value and bit pointer parameters are transferred between the fieldbus adapter 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** | **1…29** | **0** |
| **Name** | Source type | 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 |
| **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 | |
| 01.14 | Motor speed est | REAL | 32 | -30000…30000 | rpm | 2 ms | |
| 01.15 | Temp inverter | 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 | |
| 01.21 | 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 | 0b000000…0b111111 | - | 2 ms | |
| 02.02 | RO status | Pb | 16 | 0b00000…0b11111 | - | 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 | |
| 02.05 | AI1 scaled | REAL | 32 | -32768…32768 | - | 2 ms | |
| 02.06 | AI2 | REAL | 16 | -11…11 V or -22…22 mA | V or mA | 2 ms | |
| 02.07 | AI2 scaled | REAL | 32 | -32768…32768 | - | 2 ms | |
| 02.08 | AI3 | REAL | 16 | -22…22 | mA | 2 ms | |
| 02.09 | AI3 scaled | REAL | 32 | -32768…32768 | - | 2 ms | |
| 02.10 | AI4 | REAL | 16 | -22…22 | mA | 2 ms | |
| 02.11 | AI4 scaled | REAL | 32 | -32768…32768 | - | 2 ms | |
| 02.12 | AI5 | REAL | 16 | -22…22 | mA | 2 ms | |
| 02.13 | AI5 scaled | REAL | 32 | -32768…32768 | - | 2 ms | |

| No. | Name | Type | Data length | Range | Unit | Update time | Notes |
|---|---|---|---|---|---|---|---|
| *02.16* | *AO1* | REAL | 16 | 0 … 22.7 | mA | 2 ms | |
| *02.17* | *AO2* | REAL | 16 | 0 … 22.7 | mA | 2 ms | |
| *02.18* | *AO3* | REAL | 16 | 0 … 22.7 | mA | 2 ms | |
| *02.19* | *AO4* | REAL | 16 | 0 … 22.7 | mA | 2 ms | |
| *02.20* | *Freq in* | REAL | 32 | -32768…32768 | - | 250 µs | |
| *02.21* | *Freq out* | REAL | 32 | 0…32767 | Hz | 250 µs | |
| *02.22* | *FBA main cw* | Pb | 32 | 0x00000000 … 0xFFFFFFFF | - | 500 µs | |
| *02.24* | *FBA main sw* | Pb | 32 | 0x00000000 … 0xFFFFFFFF | - | 500 µs | |
| *02.26* | *FBA main ref1* | INT32 | 32 | -2147483647 … 2147483647 | - | 500 µs | |
| *02.27* | *FBA main ref2* | INT32 | 32 | -2147483647 … 2147483647 | - | 500 µs | |
| *02.34* | *Panel ref* | REAL | 32 | -32768…32768 | rpm or % | 10 ms | |
| *02.36* | *EFB main cw* | Pb | 32 | 0x00000000 … 0xFFFFFFFF | - | 10 ms | |
| *02.37* | *EFB main sw* | Pb | 32 | 0x00000000 … 0xFFFFFFFF | - | 10 ms | |
| *02.38* | *EFB main ref1* | INT32 | 32 | -2147483647 … 2147483647 | - | 10 ms | |
| *02.39* | *EFB main ref2* | INT32 | 32 | -2147483647 … 2147483647 | - | 10 ms | |
| *02.40* | *FBA setpoint* | REAL | 32 | 0 … 32768 | % | - | |
| *02.41* | *FBA act val* | REAL | 32 | 0 … 32768 | % | - | |
| *02.42* | *Shared DI* | Pb | 32 | 0x00000000 … 0xFFFFFFFF | - | 10 ms | |
| *02.43* | *Shared signal 1* | REAL | 32 | 0…32767 | - | 10 ms | |
| *02.44* | *Shared signal 2* | REAL | 32 | 0…32767 | - | 10 ms | |
| ***03 Control values*** | | | | | | | |
| *03.03* | *SpeedRef unramp* | REAL | 32 | -30000…30000 | rpm | 250 µs | |
| *03.05* | *SpeedRef ramped* | REAL | 32 | -30000…30000 | rpm | 250 µs | |
| *03.06* | *SpeedRef used* | REAL | 32 | -30000…30000 | rpm | 250 µs | |
| *03.07* | *Speed error filt* | REAL | 32 | -30000…30000 | rpm | 250 µs | |
| *03.08* | *Acc comp torq* | REAL | 16 | -1600…1600 | % | 250 µs | |
| *03.09* | *Torq ref sp ctrl* | REAL | 16 | -1600…1600 | % | 250 µs | |
| *03.13* | *Torq ref to TC* | REAL | 16 | -1600…1600 | % | 250 µs | |
| *03.14* | *Torq ref used* | REAL | 16 | -1600…1600 | % | 250 µs | |
| *03.17* | *Flux actual* | REAL24 | 16 | 0…200 | % | 2 ms | |
| *03.20* | *Max speed ref* | REAL | 16 | 0…30000 | rpm | 2 ms | |
| *03.21* | *Min speed ref* | REAL | 16 | -30000…0 | rpm | 2 ms | |
| ***04 Appl values*** | | | | | | | |
| *04.01* | *Act val* | REAL | 32 | 0…32768 | % | 2 ms | |
| *04.02* | *Setpoint* | REAL | 32 | 0…32768 | % | 2 ms | |
| *04.04* | *Process PID err* | REAL | 32 | -32768…32768 | - | 2 ms | |
| *04.05* | *Process PID out* | REAL | 32 | -32768…32768 | - | 2 ms | |
| *04.06* | *Process var1* | REAL | 32 | -32768…32768 | - | 10 ms | |
| *04.07* | *Process var2* | REAL | 32 | -32768…32768 | - | 10 ms | |
| *04.08* | *Process var3* | REAL | 32 | -32768…32768 | - | 10 ms | |
| *04.09* | *Counter ontime1* | UINT32 | 32 | 0…2147483647 | s | 10 ms | |
| *04.10* | *Counter ontime2* | UINT32 | 32 | 0…2147483647 | s | 10 ms | |
| *04.11* | *Counter edge1* | UINT32 | 32 | 0…2147483647 | - | 10 ms | |

| No. | Name | Type | Data length | Range | Unit | Update time | Notes |
|---|---|---|---|---|---|---|---|
| 04.12 | Counter edge2 | UINT32 | 32 | 0…2147483647 | - | 10 ms | |
| 04.13 | Counter value1 | UINT32 | 32 | 0…2147483647 | - | 10 ms | |
| 04.14 | Counter value2 | UINT32 | 32 | 0…2147483647 | - | 10 ms | |
| 04.20 | Act val 1 out | REAL | 32 | 0…32768 | % | 10 ms | |
| 04.21 | Act val 2 out | REAL | 32 | 0…32768 | % | 10 ms | |
| 04.22 | Act val % | REAL | 16 | 0…100 | % | 10 ms | |
| 04.23 | Setpoint val 1 | REAL | 32 | 0…32768 | % | 10 ms | |
| 04.24 | Setpoint val 2 | REAL | 32 | 0…32768 | % | 10 ms | |
| 04.25 | Setpoint val % | REAL | 16 | 0…100 | % | 10 ms | |
| 04.26 | Wake up level | REAL | 32 | -32768…32768 | - | 10 ms | |
| 04.27 | Shared source | UINT32 | 16 | 0…8 | - | 10 ms | |
| 04.28 | Pump runtime | INT32 | 32 | 0…35791394.1 | h | 10 ms | |
| 04.29 | Trad 1 runtime | INT32 | 32 | 0…35791394.1 | h | 10 ms | |
| 04.30 | Trad 2 runtime | INT32 | 32 | 0…35791394.1 | h | 10 ms | |
| 04.31 | Trad 3 runtime | INT32 | 32 | 0…35791394.1 | h | 10 ms | |
| 04.32 | Trad 4 runtime | INT32 | 32 | 0…35791394.1 | h | 10 ms | |
| 04.33 | Trad 5 runtime | INT32 | 32 | 0…35791394.1 | h | 10 ms | |
| 04.34 | Trad 6 runtime | INT32 | 32 | 0…35791394.1 | h | 10 ms | |
| 04.35 | Trad 7 runtime | INT32 | 32 | 0…35791394.1 | h | 10 ms | |
| 04.36 | Trad 8 runtime | INT32 | 32 | 0…35791394.1 | h | 10 ms | |
| **05 Pump values** | | | | | | | |
| 05.01 | MF status | UINT32 | 16 | 0…3 | - | 2 ms | |
| 05.02 | Trad pump cmd | Pb | 16 | 0b00000000 … 0b11111111 | - | 10 ms | |
| 05.03 | Trad master | UINT32 | 16 | 0…8 | - | 10 ms | |
| 05.04 | Nbr aux pumps on | INT32 | 32 | 0…65535 | - | 10 ms | |
| 05.05 | Flow act | REAL | 32 | 0…32767 | $m^3/s$ | 10 ms | |
| 05.06 | Flow by head | REAL | 32 | 0…32767 | $m^3/s$ | 10 ms | |
| 05.07 | Flow by power | REAL | 32 | 0…32767 | $m^3/s$ | 10 ms | |
| 05.08 | Total flow | UINT32 | 32 | 0…2147483647 | $m^3$ | 10 ms | |
| 05.09 | Bypass ref | REAL | 32 | -32768…32768 | rpm | 10 ms | |
| 05.10 | Speed ref | REAL | 32 | -32768…32767 | rpm | 10 ms | |
| 05.20 | kWh current read | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.21 | kWh prev read | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.22 | kWh posprev read | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.23 | kWh cur mon read | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.24 | kWh January | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.25 | kWh February | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.26 | kWh March | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.27 | kWh April | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.28 | kWh May | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.29 | kWh June | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.30 | kWh July | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.31 | kWh August | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.32 | kWh September | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.33 | kWh October | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.34 | kWh November | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.35 | kWh December | UINT32 | 32 | 0…2147483647 | kWh | 10 ms | |
| 05.36 | First in order | UINT32 | 16 | 0…8 | - | 10 ms | |
| 05.37 | Time autochg | UINT32 | 32 | 0…2147483647 | h | 10 ms | |
| 05.39 | Next start node | UINT32 | 16 | 0…8 | - | 10 ms | |

| No. | Name | Type | Data length | Range | Unit | Update time | Notes |
|---|---|---|---|---|---|---|---|
| ***06 Drive status*** | | | | | | | |
| *06.01* | *Status word1* | Pb | 16 | 0x0000…0xFFFF | - | 2 ms | |
| *06.02* | *Status word2* | Pb | 16 | 0x0000…0xFFFF | - | 2 ms | |
| *06.03* | *Speed ctrl stat* | Pb | 16 | 0x0000…0xFFFF | - | 250 µs | |
| *06.05* | *Limit word1* | Pb | 16 | 0x0000…0xFFFF | - | 250 µs | |
| *06.07* | *Torq lim status* | Pb | 16 | 0x0000…0xFFFF | - | 250 µs | |
| *06.12* | *Op mode ack* | enum | 16 | 0…11 | - | 2 ms | |
| *06.13* | *Superv status* | Pb | 16 | 0b000…0b111 | - | 2 ms | |
| *06.14* | *Timed func stat* | Pb | 16 | 0b00000…0b11111 | - | 10 ms | |
| *06.15* | *Counter status* | Pb | 16 | 0b000000…0b111111 | - | 10 ms | |
| *06.20* | *Pump status word* | Pb | 16 | 0x00000000 … 0xFFFFFFFF | - | 2 ms | |
| *06.21* | *Level status* | Pb | 16 | 0x00000000 … 0xFFFFFFFF | - | 10 ms | |
| *06.22* | *MF status word* | Pb | 16 | 0x00000000 … 0xFFFFFFFF | - | 2 ms | |
| ***08 Alarms & faults*** | | | | | | | |
| *08.01* | *Active fault* | enum | 16 | 0…65535 | - | - | |
| *08.02* | *Last fault* | enum | 16 | 0…2147483647 | - | - | |
| *08.03* | *Fault time hi* | INT32 | 32 | $-2^{31}…2^{31}$ - 1 | (date) | - | |
| *08.04* | *Fault time lo* | INT32 | 32 | 00:00:00 … 24:00:00 | (time) | - | |
| *08.05* | *Alarm word1* | UINT32 | 16 | 0x0000…0xFFFF | - | 2 ms | |
| *08.06* | *Alarm word2* | UINT32 | 16 | 0x0000…0xFFFF | - | 2 ms | |
| *08.07* | *Alarm word3* | UINT32 | 16 | 0x0000…0xFFFF | - | 2 ms | |
| *08.08* | *Alarm word4* | UINT32 | 16 | 0x0000…0xFFFF | - | 2 ms | |
| *08.09* | *Alarm word5* | UINT32 | 16 | 0x0000…0xFFFF | - | 2 ms | |
| *08.10* | *Alarm word6* | UINT32 | 16 | 0x0000…0xFFFF | - | 2 ms | |
| *08.20* | *Pump fault word* | Pb | 16 | 0x0000…0xFFFF | - | 2 ms | |
| *08.21* | *Pump alarm word* | Pb | 16 | 0x00000000 … 0xFFFFFFFF | - | 2 ms | |
| ***09 System info*** | | | | | | | |
| *09.01* | *Drive type* | INT32 | 16 | - | - | - | |
| *09.02* | *Drive rating ID* | INT32 | 16 | 0…65535 | - | - | |
| *09.03* | *Firmware ID* | Pb | 16 | - | - | - | |
| *09.04* | *Firmware ver* | Pb | 16 | - | - | - | |
| *09.05* | *Firmware patch* | Pb | 16 | - | - | - | |
| *09.10* | *Int logic ver* | Pb | 32 | - | - | - | |
| *09.20* | *Option slot1* | INT32 | 16 | 0…25 | - | - | |
| *09.21* | *Option slot2* | INT32 | 16 | 0…25 | - | - | |

# Parameter groups 10…99

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| ***10 Start/stop/dir*** | | | | | | |
| *10.01* | *Ext1 start func* | enum | 16 | 0…6 | - | *In1* |
| *10.02* | *Ext1 start in1* | Bit pointer | 32 | - | - | *DI1* |
| *10.03* | *Ext1 start in2* | Bit pointer | 32 | - | - | C.FALSE |
| *10.04* | *Ext2 start func* | enum | 16 | 0…6 | - | *In1* |
| *10.05* | *Ext2 start in1* | Bit pointer | 32 | - | - | *DI1* |
| *10.06* | *Ext2 start in2* | Bit pointer | 32 | - | - | C.FALSE |
| *10.10* | *Fault reset sel* | Bit pointer | 32 | - | - | *DI3* |
| *10.11* | *Run enable* | Bit pointer | 32 | - | - | C.TRUE |
| *10.13* | *Em stop off3* | Bit pointer | 32 | - | - | C.TRUE |
| *10.15* | *Em stop off1* | Bit pointer | 32 | - | - | C.TRUE |
| *10.17* | *Start enable* | Bit pointer | 32 | - | - | C.TRUE |
| *10.19* | *Start inhibit* | enum | 16 | 0…1 | - | *Disabled* |
| *10.20* | *Start intrl func* | enum | 16 | 0…1 | - | *Off3 stop* |
| ***11 Start/stop mode*** | | | | | | |
| *11.01* | *Start mode* | enum | 16 | 0…2 | - | *Automatic* |
| *11.02* | *Dc-magn time* | UINT32 | 16 | 0…10000 | ms | 500 ms |
| *11.03* | *Stop mode* | enum | 16 | 1…2 | - | *Coast* |
| *11.04* | *Dc hold speed* | REAL | 16 | 0…1000 | rpm | 5.0 rpm |
| *11.05* | *Dc hold curr ref* | UINT32 | 16 | 0…100 | % | 30% |
| *11.06* | *Dc hold* | enum | 16 | 0…1 | - | *Disabled* |
| ***12 Operating mode*** | | | | | | |
| *12.01* | *Ext1/Ext2 sel* | Bit pointer | 32 | - | - | *DI5* |
| *12.05* | *Ext2 ctrl mode* | enum | 16 | 1…2 | - | *PID* |
| ***13 Analogue inputs*** | | | | | | |
| *13.01* | *AI1 filt time* | REAL | 16 | 0…30 | s | 0.100 s |
| *13.02* | *AI1 max* | REAL | 16 | -22…22 mA or -11…11 V | mA or V | 20.000 mA or 10.000 V |
| *13.03* | *AI1 min* | REAL | 16 | -22…22 mA or -11…11 V | mA or V | 4.000 mA or 2.000 V |
| *13.04* | *AI1 max scale* | REAL | 32 | -32768…32768 | - | 1500.000 |
| *13.05* | *AI1 min scale* | REAL | 32 | -32768…32768 | - | 0.000 |
| *13.06* | *AI2 filt time* | REAL | 16 | 0…30 | s | 0.100 s |
| *13.07* | *AI2 max* | REAL | 16 | -22…22 mA or -11…11 V | mA or V | 20.000 mA or 10.000 V |
| *13.08* | *AI2 min* | REAL | 16 | -22…22 mA or -11…11 V | mA or V | 4.000 mA or 2.000 V |
| *13.09* | *AI2 max scale* | REAL | 32 | -32768…32768 | - | 100.000 |
| *13.10* | *AI2 min scale* | REAL | 32 | -32768…32768 | - | 0.000 |
| *13.11* | *AI3 filt time* | REAL | 16 | 0…30 | s | 0.100 s |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| *13.12* | *AI3 max* | REAL | 16 | -22…22 mA or -11…11 V | mA or V | 22.000 mA or 10.000 V |
| *13.13* | *AI3 min* | REAL | 16 | -22…22 mA or -11…11 V | mA or V | 4.000 mA or 2.000 V |
| *13.14* | *AI3 max scale* | REAL | 32 | -32768…32768 | - | 1500.000 |
| *13.15* | *AI3 min scale* | REAL | 32 | -32768…32768 | - | 0.000 |
| *13.16* | *AI4 filt time* | REAL | 16 | 0…30 | s | 0.100 s |
| *13.17* | *AI4 max* | REAL | 16 | -22…22 mA or -11…11 V | mA or V | 22.000 mA or 10.000 V |
| *13.18* | *AI4 min* | REAL | 16 | -22…22 mA or -11…11 V | mA or V | 4.000 mA or 2.000 V |
| *13.19* | *AI4 max scale* | REAL | 32 | -32768…32768 | - | 1500.000 |
| *13.20* | *AI4 min scale* | REAL | 32 | -32768…32768 | - | 0.000 |
| *13.21* | *AI5 filt time* | REAL | 16 | 0…30 | s | 0.100 s |
| *13.22* | *AI5 max* | REAL | 16 | -22…22 mA or -11…11 V | mA or V | 22.000 mA or 10.000 V |
| *13.23* | *AI5 min* | REAL | 16 | -22…22 mA or -11…11 V | mA or V | 4.000 mA or 2.000 V |
| *13.24* | *AI5 max scale* | REAL | 32 | -32768…32768 | - | 1500.000 |
| *13.25* | *AI5 min scale* | REAL | 32 | -32768…32768 | - | 0.000 |
| *13.31* | *AI tune* | enum | 16 | 0…4 | - | *No action* |
| *13.32* | *AI superv func* | enum | 16 | 0…3 | - | *No* |
| *13.33* | *AI superv cw* | UINT32 | 32 | 0b0000…0b1111 | - | 0b0000 |
| ***14 Digital I/O*** | | | | | | |
| *14.01* | *DI invert mask* | Pb | 16 | 0b00000 … 0b11111 | - | 0b00000 |
| *14.02* | *DIO1 conf* | enum | 16 | 0…2 | - | *Output* |
| *14.03* | *DIO1 out src* | Bit pointer | 32 | - | - | *Ready* |
| *14.04* | *DIO1 Ton* | UINT32 | 16 | 0…3000 | s | 0.0 s |
| *14.05* | *DIO1 Toff* | UINT32 | 16 | 0…3000 | s | 0.0 s |
| *14.06* | *DIO2 conf* | enum | 16 | 0…2 | - | *Output* |
| *14.07* | *DIO2 out src* | Bit pointer | 32 | - | - | *Running* |
| *14.08* | *DIO2 Ton* | UINT32 | 16 | 0…3000 | s | 0.0 s |
| *14.09* | *DIO2 Toff* | UINT32 | 16 | 0…3000 | s | 0.0 s |
| *14.10* | *DIO3 conf* | enum | 16 | 0…1 | - | *Output* |
| *14.11* | *DIO3 out src* | Bit pointer | 32 | - | - | *Fault(-1)* |
| *14.14* | *DIO4 conf* | enum | 16 | 0…1 | - | *Output* |
| *14.15* | *DIO4 out src* | Bit pointer | 32 | - | - | *Ready relay* |
| *14.42* | *RO1 src* | Bit pointer | 32 | - | - | *Ready* |
| *14.43* | *RO1 Ton* | UINT32 | 16 | 0…3000 | s | 0.0 s |
| *14.44* | *RO1 Toff* | UINT32 | 16 | 0…3000 | s | 0.0 s |
| *14.45* | *RO2 src* | Bit pointer | 32 | - | - | *Fault(-1)* |
| *14.48* | *RO3 src* | Bit pointer | 32 | - | - | *Ready relay* |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 14.51 | RO4 src | Bit pointer | 32 | - | - | Ref running |
| 14.54 | RO5 src | Bit pointer | 32 | - | - | Ref running |
| 14.57 | Freq in max | REAL | 16 | 3…32768 | Hz | 1000 Hz |
| 14.58 | Freq in min | REAL | 16 | 3…32768 | Hz | 3 Hz |
| 14.59 | Freq in max scal | REAL | 16 | -32768…32768 | - | 1500 |
| 14.60 | Freq in min scal | REAL | 16 | -32768…32768 | - | 0 |
| 14.61 | Freq out src | Val pointer | 32 | - | - | P.01.01 |
| 14.62 | Freq out max src | REAL | 16 | 0…32768 | - | 1500 |
| 14.63 | Freq out min src | REAL | 16 | 0…32768 | - | 0 |
| 14.64 | Freq out max sca | REAL | 16 | 3…32768 | Hz | 1000 Hz |
| 14.65 | Freq out min sca | REAL | 16 | 3…32768 | Hz | 3 Hz |
| 14.66 | RO6 src | Bit pointer | 32 | - | - | Ref running |
| 14.72 | DIO invert mask | Pb | 16 | 0b0000000000 … 0b1111111111 | - | 0b0000000000 |
| **15 Analogue outputs** | | | | | | |
| 15.01 | AO1 src | Val pointer | 32 | - | - | Current |
| 15.02 | AO1 filt time | REAL | 16 | 0…30 | s | 0.100 s |
| 15.03 | AO1 out max | REAL | 16 | 0 … 22.7 | mA | 20.000 mA |
| 15.04 | AO1 out min | REAL | 16 | 0 … 22.7 | mA | 4.000 mA |
| 15.05 | AO1 src max | REAL | 32 | -32768…32768 | - | 100.000 |
| 15.06 | AO1 src min | REAL | 32 | -32768…32768 | - | 0.000 |
| 15.07 | AO2 src | Val pointer | 32 | - | - | Speed rpm |
| 15.08 | AO2 filt time | REAL | 16 | 0…30 | s | 0.100 s |
| 15.09 | AO2 out max | REAL | 16 | 0…22.7 | mA | 20.000 mA |
| 15.10 | AO2 out min | REAL | 16 | 0…22.7 | mA | 4.000 mA |
| 15.11 | AO2 src max | REAL | 32 | -32768…32768 | - | 100.000 |
| 15.12 | AO2 src min | REAL | 32 | -32768…32768 | - | 0.000 |
| 15.13 | AO3 src | Val pointer | 32 | - | - | Frequency |
| 15.14 | AO3 filt time | REAL | 16 | 0…30 | s | 0.100 s |
| 15.15 | AO3 out max | REAL | 16 | 0 … 22.7 | mA | 22.000 mA |
| 15.16 | AO3 out min | REAL | 16 | 0 … 22.7 | mA | 4.000 mA |
| 15.17 | AO3 src max | REAL | 32 | -32768…32768 | - | 50.000 |
| 15.18 | AO3 src min | REAL | 32 | -32768…32768 | - | 0.000 |
| 15.25 | AO ctrl word | UINT32 | 32 | 0b00…0b11 | - | 0b00 |
| **16 System** | | | | | | |
| 16.01 | Local lock | Bit pointer | 32 | - | - | C.FALSE |
| 16.02 | Parameter lock | enum | 16 | 0…2 | - | Open |
| 16.03 | Pass code | INT32 | 32 | 0…2147483647 | - | 0 |
| 16.04 | Param restore | enum | 16 | 0…2 | - | Done |
| 16.07 | Param save | enum | 16 | 0…1 | - | Done |
| 16.09 | User set sel | enum | 32 | 1…10 | - | No request |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 16.10 | User set log | Pb | 32 | 0…4294967295 | - | N/A |
| 16.11 | User IO sel lo | Bit pointer | 32 | - | - | C.FALSE |
| 16.12 | User IO sel hi | Bit pointer | 32 | - | - | C.FALSE |
| 16.14 | Reset ChgParLog | enum | 16 | 0…1 | - | Done |
| 16.16 | Menu set active | enum | 16 | 0…32 | - | Single short |
| 16.17 | Power unit | enum | 16 | 0…1 | - | kW |
| 16.20 | Macro selected | enum | 16 | 0…6 | - | Factory def |
| 16.21 | Menu selection | enum | 16 | 0…2 | - | Short |
| **19 Speed calculation** | | | | | | |
| 19.01 | Speed scaling | REAL | 16 | 0…30000 | rpm | 1500 rpm |
| 19.02 | Speed fb sel | enum | 16 | - | - | Estimated |
| 19.03 | MotorSpeed filt | REAL | 32 | 0…10000 | ms | 8.000 ms |
| 19.06 | Zero speed limit | REAL | 32 | 0…30000 | rpm | 30.00 rpm |
| 19.07 | Zero speed delay | UINT32 | 16 | 0…30000 | ms | 0 ms |
| 19.08 | Above speed lim | REAL | 16 | 0…30000 | rpm | 0 rpm |
| 19.09 | Speed TripMargin | REAL | 32 | 0…10000 | rpm | 500.0 rpm |
| 19.10 | Speed window | REAL | 16 | 0…30000 | rpm | 100 rpm |
| **20 Limits** | | | | | | |
| 20.01 | Maximum speed | REAL | 32 | 0…30000 | rpm | 1500 rpm |
| 20.02 | Minimum speed | REAL | 32 | -30000…0 | rpm | 0 rpm |
| 20.03 | Pos speed ena | Bit pointer | 32 | - | - | C.TRUE |
| 20.04 | Neg speed ena | Bit pointer | 32 | - | - | C.FALSE |
| 20.05 | Maximum current | REAL | 32 | 0…30000 | A | 0.00 A |
| 20.06 | Torq lim sel | Bit pointer | 32 | - | - | C.FALSE |
| 20.07 | Maximum torque1 | REAL | 16 | 0…1600 | % | 300.0% |
| 20.08 | Minimum torque1 | REAL | 16 | -1600…0 | % | -300.0% |
| 20.09 | Maximum torque2 | REAL | 16 | - | - | Max torque1 |
| 20.10 | Minimum torque2 | REAL | 16 | - | - | Min torque1 |
| 20.12 | P motoring lim | REAL | 16 | 0…1600 | % | 300.0% |
| 20.13 | P generating lim | REAL | 16 | 0…1600 | % | 300.0% |
| **21 Speed ref** | | | | | | |
| 21.01 | Speed ref1 sel | Val pointer | 32 | - | - | AI1 scaled |
| 21.02 | Speed ref2 sel | Val pointer | 32 | - | - | Zero |
| 21.05 | Speed share | REAL | 16 | -8…8 | - | 1.000 |
| 21.09 | SpeedRef min abs | REAL | 16 | 0…30000 | rpm | 0 rpm |
| **22 Speed ref ramp** | | | | | | |
| 22.02 | Acc time | REAL | 32 | 0…1800 | s | 5.000 s |
| 22.03 | Dec time | REAL | 32 | 0…1800 | s | 5.000 s |
| 22.06 | Shape time acc1 | REAL | 32 | 0…1000 | s | 0.100 s |
| 22.07 | Shape time acc2 | REAL | 32 | 0…1000 | s | 0.100 s |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 22.08 | Shape time dec1 | REAL | 32 | 0…1000 | s | 0.100 s |
| 22.09 | Shape time dec2 | REAL | 32 | 0…1000 | s | 0.100 s |
| 22.12 | Em stop time | REAL | 32 | 0…1800 | s | 3.000 s |
| **23 Speed ctrl** | | | | | | |
| 23.01 | Proport gain | REAL | 16 | 0…200 | - | 10.00 |
| 23.02 | Integration time | REAL | 32 | 0…600 | s | 0.500 s |
| 23.03 | Derivation time | REAL | 16 | 0…10 | s | 0.000 s |
| 23.04 | Deriv filt time | REAL | 16 | 0…1000 | ms | 8.0 ms |
| 23.05 | Acc comp DerTime | REAL | 32 | 0…600 | s | 0.00 s |
| 23.06 | Acc comp Ftime | REAL | 16 | 0…1000 | ms | 8.0 ms |
| 23.07 | Speed err Ftime | REAL | 16 | 0…1000 | ms | 0.0 ms |
| 23.08 | Speed additive | Val pointer | 32 | - | - | Zero |
| 23.09 | Max torq sp ctrl | REAL | 16 | -1600…1600 | % | 300.0% |
| 23.10 | Min torq sp ctrl | REAL | 16 | -1600…1600 | % | -300.0% |
| 23.11 | SpeedErr winFunc | enum | 16 | 0…2 | - | Disabled |
| 23.12 | SpeedErr win hi | REAL | 16 | 0…3000 | rpm | 0 rpm |
| 23.13 | SpeedErr win lo | REAL | 16 | 0…3000 | rpm | 0 rpm |
| 23.14 | Drooping rate | REAL | 16 | 0…100 | % | 0.00% |
| 23.15 | PI adapt max sp | REAL | 16 | 0…30000 | rpm | 0 rpm |
| 23.16 | PI adapt min sp | REAL | 16 | 0…30000 | rpm | 0 rpm |
| 23.17 | Pcoef at min sp | REAL | 16 | 0…10 | - | 1.000 |
| 23.18 | Icoef at min sp | REAL | 16 | 0…10 | - | 1.000 |
| 23.20 | PI tune mode | enum | 16 | 0…4 | - | Done |
| 23.21 | Tune bandwidth | REAL | 16 | 0…2000 | Hz | 100.00 Hz |
| 23.22 | Tune damping | REAL | 16 | 0…200 | - | 0.5 |
| **25 Critical speed** | | | | | | |
| 25.01 | Crit speed sel | enum | 16 | 0…1 | - | Disable |
| 25.02 | Crit speed1 lo | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| 25.03 | Crit speed1 hi | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| 25.04 | Crit speed2 lo | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| 25.05 | Crit speed2 hi | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| 25.06 | Crit speed3 lo | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| 25.07 | Crit speed3 hi | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| **26 Constant speeds** | | | | | | |
| 26.01 | Const speed func | Pb | 16 | 0b00…0b11 | - | 0b00 |
| 26.02 | Const speed sel1 | Bit pointer | 32 | - | - | DI2 |
| 26.03 | Const speed sel2 | Bit pointer | 32 | - | - | C.FALSE |
| 26.04 | Const speed sel3 | Bit pointer | 32 | - | - | C.FALSE |
| 26.06 | Const speed1 | REAL | 16 | -30000…30000 | rpm | 1200 rpm |
| 26.07 | Const speed2 | REAL | 16 | -30000…30000 | rpm | 0 rpm |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| *26.08* | *Const speed3* | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| *26.09* | *Const speed4* | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| *26.10* | *Const speed5* | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| *26.11* | *Const speed6* | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| *26.12* | *Const speed7* | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| ***27 Process PID*** | | | | | | |
| *27.01* | *PID setpoint sel* | Val pointer | 32 | - | - | *Setpoint %* |
| *27.12* | *PID gain* | REAL | 16 | 0…100 | - | 1.00 |
| *27.13* | *PID integ time* | REAL | 16 | 0…320 | s | 60.00 s |
| *27.14* | *PID deriv time* | REAL | 16 | 0…10 | s | 0.00 s |
| *27.15* | *PID deriv filter* | REAL | 16 | 0…10 | s | 1.00 s |
| *27.16* | *PID error inv* | Bit pointer | 32 | - | - | C.FALSE |
| *27.18* | *PID maximum* | REAL | 32 | -32768…32768 | - | 100.0 |
| *27.19* | *PID minimum* | REAL | 32 | -32768…32768 | - | 0.0 |
| *27.30* | *Pid ref freeze* | Bit pointer | 32 | - | - | *No* |
| *27.31* | *Pid out freeze* | Bit pointer | 32 | - | - | *No* |
| *27.32* | *Pipefill ref acc* | REAL | 16 | 0…100 | s | 5 s |
| *27.33* | *Pipefill ref dec* | REAL | 16 | 0…100 | s | 5 s |
| *27.34* | *PID bal ena* | Val pointer | 32 | - | - | C.FALSE |
| *27.35* | *PID bal ref* | REAL | 32 | -32768…32768 | % | 0.0% |
| *27.36* | *Pump scal speed* | Val pointer | 32 | - | - | *Speed scal* |
| ***28 Procact sel*** | | | | | | |
| *28.01* | *Act val 1/2 sel* | Bit pointer | 32 | - | - | *Act val 1* |
| *28.02* | *Act val 1 src* | Val pointer | 32 | - | - | *AI2 scaled* |
| *28.03* | *Act val 2 src* | Val pointer | 32 | - | - | *Zero* |
| *28.04* | *Act val func* | enum | 16 | 0…8 | - | *Act1* |
| *28.05* | *Act max val* | REAL | 32 | 0…32768 | % | 100.00% |
| *28.06* | *Act unit sel* | enum | 32 | 0…32767 | - | *%* |
| *28.07* | *Act FBA scaling* | enum | 16 | 0…3 | - | *Src/100* |
| ***29 Setpoint sel*** | | | | | | |
| *29.01* | *Setpoint 1 / 2 sel* | Bit pointer | 32 | - | - | *Setpoint 1* |
| *29.02* | *Setpoint 1 src* | Val pointer | 32 | - | - | *Int set 1* |
| *29.03* | *Setpoint 2 src* | Val pointer | 32 | - | - | *Zero* |
| *29.04* | *Internal set 1* | REAL | 32 | 0…32768 | % | 40.00% |
| *29.05* | *Internal set 2* | REAL | 32 | 0…32768 | % | 60.00% |
| *29.06* | *Reference step 1* | REAL | 16 | 0…100 | % | 0.00% |
| *29.07* | *Reference step 2* | REAL | 16 | 0…100 | % | 0.00% |
| *29.08* | *Reference step 3* | REAL | 16 | 0…100 | % | 0.00% |
| *29.09* | *Reference step 4* | REAL | 16 | 0…100 | % | 0.00% |
| *29.10* | *Reference step 5* | REAL | 16 | 0…100 | % | 0.00% |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 29.11 | *Reference step 6* | REAL | 16 | 0…100 | % | 0.00% |
| 29.12 | *Reference step 7* | REAL | 16 | 0…100 | % | 0.00% |
| ***30 Fault functions*** | | | | | | |
| 30.01 | *External fault* | Bit pointer | 32 | - | - | C.TRUE |
| 30.02 | *Speed ref safe* | REAL | 16 | -30000…30000 | rpm | 0 rpm |
| 30.03 | *Local ctrl loss* | enum | 16 | 0…3 | - | *Fault* |
| 30.04 | *Mot phase loss* | enum | 16 | 0…1 | - | *Fault* |
| 30.05 | *Earth fault* | enum | 16 | 0…2 | - | *Fault* |
| 30.06 | *Suppl phs loss* | enum | 16 | 0…1 | - | *Fault* |
| 30.07 | *Sto diagnostic* | enum | 16 | 1…4 | - | *Fault* |
| 30.08 | *Cross connection* | enum | 16 | 0…1 | - | *Fault* |
| 30.09 | *Stall function* | Pb | 16 | 0b000…0b111 | - | 0b111 |
| 30.10 | *Stall curr lim* | REAL | 16 | 0…1600 | % | 200.0% |
| 30.11 | *Stall freq hi* | REAL | 16 | 0.5 … 1000 | Hz | 15.0 Hz |
| 30.12 | *Stall time* | UINT32 | 16 | 0…3600 | s | 20 s |
| ***31 Motor therm prot*** | | | | | | |
| 31.01 | *Mot temp1 prot* | enum | 16 | 0…2 | - | *No* |
| 31.02 | *Mot temp1 src* | enum | 16 | 0…12 | - | *Estimated* |
| 31.03 | *Mot temp1 almLim* | INT32 | 16 | 0…200 | °C | 90 °C |
| 31.04 | *Mot temp1 fltLim* | INT32 | 16 | 0…200 | °C | 110 °C |
| 31.05 | *Mot temp2 prot* | enum | 16 | 0…2 | - | *No* |
| 31.06 | *Mot temp2 src* | enum | 16 | 0…12 | - | *Estimated* |
| 31.07 | *Mot temp2 almLim* | INT32 | 16 | 0…200 | °C | 90 °C |
| 31.08 | *Mot temp2 fltLim* | INT32 | 16 | 0…200 | °C | 110 °C |
| 31.09 | *Mot ambient temp* | INT32 | 16 | -60…100 | °C | 20 °C |
| 31.10 | *Mot load curve* | INT32 | 16 | 50…150 | % | 100% |
| 31.11 | *Zero speed load* | INT32 | 16 | 50…150 | % | 100% |
| 31.12 | *Break point* | INT32 | 16 | 0.01…500 | Hz | 45.00 Hz |
| 31.13 | *Mot nom tempRise* | INT32 | 16 | 0…300 | °C | 80 °C |
| 31.14 | *Mot therm time* | INT32 | 16 | 100…10000 | s | 256 s |
| ***32 Automatic reset*** | | | | | | |
| 32.01 | *Autoreset sel* | Pb | 16 | 0b000000…0b111111 | - | 0b000000 |
| 32.02 | *Number of trials* | UINT32 | 16 | 0…5 | - | 0 |
| 32.03 | *Trial time* | UINT32 | 16 | 1…600 | s | 30.0 s |
| 32.04 | *Delay time* | UINT32 | 16 | 0…120 | s | 0.0 s |
| ***33 Supervision*** | | | | | | |
| 33.01 | *Superv1 func* | enum | 16 | 0…4 | - | *Disabled* |
| 33.02 | *Superv1 act* | Val pointer | 32 | - | - | *Speed rpm* |
| 33.03 | *Superv1 hi* | REAL | 32 | -32768…32768 | - | 0.00 |
| 33.04 | *Superv1 lo* | REAL | 32 | -32768…32768 | - | 0.00 |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 33.05 | Superv2 func | enum | 16 | 0…4 | - | Disabled |
| 33.06 | Superv2 act | Val pointer | 32 | - | - | Current |
| 33.07 | Superv2 hi | REAL | 32 | -32768…32768 | - | 0.00 |
| 33.08 | Superv2 lo | REAL | 32 | -32768…32768 | - | 0.00 |
| 33.09 | Superv3 func | enum | 16 | 0…4 | - | Disabled |
| 33.10 | Superv3 act | Val pointer | 32 | - | - | Torque |
| 33.11 | Superv3 hi | REAL | 32 | -32768…32768 | - | 0.00 |
| 33.12 | Superv3 lo | REAL | 32 | -32768…32768 | - | 0.00 |
| **34 User load curve** | | | | | | |
| 34.01 | Overload func | Pb | 16 | 0b000000…0b111111 | - | 0b000000 |
| 34.02 | Underload func | Pb | 16 | 0b0000…0b1111 | - | 0b0000 |
| 34.03 | Load freq1 | REAL | 16 | 1…500 | Hz | 5 Hz |
| 34.04 | Load freq2 | REAL | 16 | 1…500 | Hz | 25 Hz |
| 34.05 | Load freq3 | REAL | 16 | 1…500 | Hz | 43 Hz |
| 34.06 | Load freq4 | REAL | 16 | 1…500 | Hz | 50 Hz |
| 34.07 | Load freq5 | REAL | 16 | 1…500 | Hz | 500 Hz |
| 34.08 | Load low lim1 | REAL | 16 | 0…1600 | % | 10% |
| 34.09 | Load low lim2 | REAL | 16 | 0…1600 | % | 15% |
| 34.10 | Load low lim3 | REAL | 16 | 0…1600 | % | 25% |
| 34.11 | Load low lim4 | REAL | 16 | 0…1600 | % | 30% |
| 34.12 | Load low lim5 | REAL | 16 | 0…1600 | % | 30% |
| 34.13 | Load high lim1 | REAL | 16 | 0…1600 | % | 300% |
| 34.14 | Load high lim2 | REAL | 16 | 0…1600 | % | 300% |
| 34.15 | Load high lim3 | REAL | 16 | 0…1600 | % | 300% |
| 34.16 | Load high lim4 | REAL | 16 | 0…1600 | % | 300% |
| 34.17 | Load high lim5 | REAL | 16 | 0…1600 | % | 300% |
| 34.18 | Load integ time | UINT32 | 16 | 0…10000 | s | 100 s |
| 34.19 | Load cool time | UINT32 | 16 | 0…10000 | s | 20 s |
| 34.20 | Underload time | UINT32 | 16 | 0…10000 | s | 10 s |
| **35 Process variable** | | | | | | |
| 35.01 | Signal1 param | Val pointer | 32 | - | - | Speed % |
| 35.02 | Signal1 max | REAL | 32 | -32768…32768 | - | 300.000 |
| 35.03 | Signal1 min | REAL | 32 | -32768…32768 | - | -300.000 |
| 35.04 | Proc var1 dispf | enum | 16 | 0…5 | - | 3 |
| 35.05 | Proc var1 unit | enum | 16 | 0…98 | - | 4 |
| 35.06 | Proc var1 max | REAL | 32 | -32768…32768 | - | 300.000 |
| 35.07 | Proc var1 min | REAL | 32 | -32768…32768 | - | -300.000 |
| 35.08 | Signal2 param | Val pointer | 32 | - | - | Current % |
| 35.09 | Signal2 max | REAL | 32 | -32768…32768 | - | 300.000 |
| 35.10 | Signal2 min | REAL | 32 | -32768…32768 | - | -300.000 |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 35.11 | Proc var2 dispf | enum | 16 | 0…5 | - | 3 |
| 35.12 | Proc var2 unit | enum | 16 | 0…98 | - | 4 |
| 35.13 | Proc var2 max | REAL | 32 | -32768…32768 | - | 300.000 |
| 35.14 | Proc var2 min | REAL | 32 | -32768…32768 | - | -300.000 |
| 35.15 | Signal3 param | Val pointer | 32 | - | - | Torque |
| 35.16 | Signal3 max | REAL | 32 | -32768…32768 | - | 300.000 |
| 35.17 | Signal3 min | REAL | 32 | -32768…32768 | - | -300.000 |
| 35.18 | Proc var3 dispf | enum | 16 | 0…5 | - | 3 |
| 35.19 | Proc var3 unit | enum | 16 | 0…98 | - | 4 |
| 35.20 | Proc var3 max | REAL | 32 | -32768…32768 | - | 300.000 |
| 35.21 | Proc var3 min | REAL | 32 | -32768…32768 | - | -300.000 |
| **36 Timed functions** | | | | | | |
| 36.01 | Timers enable | Bit pointer | 32 | - | - | C.FALSE |
| 36.02 | Timers mode | Pb | 16 | 0b0000…0b1111 | - | 0b0000 |
| 36.03 | Start time1 | UINT32 | 32 | 00:00:00 … 24:00:00 | (time) | 00:00:00 |
| 36.04 | Stop time1 | UINT32 | 32 | 00:00:00 … 24:00:00 | (time) | 00:00:00 |
| 36.05 | Start day1 | enum | 16 | 1…7 | - | Monday |
| 36.06 | Stop day1 | enum | 16 | 1…7 | - | Monday |
| 36.07 | Start time2 | UINT32 | 32 | 00:00:00 … 24:00:00 | (time) | 00:00:00 |
| 36.08 | Stop time2 | UINT32 | 32 | 00:00:00 … 24:00:00 | (time) | 00:00:00 |
| 36.09 | Start day2 | enum | 16 | 1…7 | - | Monday |
| 36.10 | Stop day2 | enum | 16 | 1…7 | - | Monday |
| 36.11 | Start time3 | UINT32 | 32 | 00:00:00 … 24:00:00 | (time) | 00:00:00 |
| 36.12 | Stop time3 | UINT32 | 32 | 00:00:00 … 24:00:00 | (time) | 00:00:00 |
| 36.13 | Start day3 | enum | 16 | 1…7 | - | Monday |
| 36.14 | Stop day3 | enum | 16 | 1…7 | - | Monday |
| 36.15 | Start time4 | UINT32 | 32 | 00:00:00 … 24:00:00 | (time) | 00:00:00 |
| 36.16 | Stop time4 | UINT32 | 32 | 00:00:00 … 24:00:00 | (time) | 00:00:00 |
| 36.17 | Start day4 | enum | 16 | 1…7 | - | Monday |
| 36.18 | Stop day4 | enum | 16 | 1…7 | - | Monday |
| 36.19 | Boost signal | Bit pointer | 32 | - | - | C.FALSE |
| 36.20 | Boost time | UINT32 | 32 | 00:00:00 … 24:00:00 | (time) | 00:00:00 |
| 36.21 | Timed func1 | Pb | 16 | 0b00000…0b11111 | - | 0b00000 |
| 36.22 | Timed func2 | Pb | 16 | 0b00000…0b11111 | - | 0b00000 |
| 36.23 | Timed func3 | Pb | 16 | 0b00000…0b11111 | - | 0b00000 |
| 36.24 | Timed func4 | Pb | 16 | 0b00000…0b11111 | - | 0b00000 |
| **38 Flux ref** | | | | | | |
| 38.01 | Flux ref | REAL | 16 | 0…200 | % | 100% |
| 38.03 | U/f curve func | enum | 16 | 0…2 | - | Linear |
| 38.04 | U/f curve freq1 | REAL | 16 | 1…500 | % | 10% |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| *38.05* | *U/f curve freq2* | REAL | 16 | 1…500 | % | 30% |
| *38.06* | *U/f curve freq3* | REAL | 16 | 1…500 | % | 50% |
| *38.07* | *U/f curve freq4* | REAL | 16 | 1…500 | % | 70% |
| *38.08* | *U/f curve freq5* | REAL | 16 | 1…500 | % | 90% |
| *38.09* | *U/f curve volt1* | REAL | 16 | 0…200 | % | 20% |
| *38.10* | *U/f curve volt2* | REAL | 16 | 0…200 | % | 40% |
| *38.11* | *U/f curve volt3* | REAL | 16 | 0…200 | % | 60% |
| *38.12* | *U/f curve volt4* | REAL | 16 | 0…200 | % | 80% |
| *38.13* | *U/f curve volt5* | REAL | 16 | 0…200 | % | 100% |
| *38.16* | *Flux ref pointer* | Val pointer | 32 | - | - | P.38.01 |
| ***40 Motor control*** | | | | | | |
| *40.01* | *Motor noise* | enum | 16 | 0…2 | - | *Cyclic* |
| *40.03* | *Slip gain* | REAL24 | 32 | 0…200 | % | 100% |
| *40.04* | *Voltage reserve* | REAL24 | 32 | -4…50 | % | -2% |
| *40.07* | *IR-compensation* | REAL24 | 32 | 0…50 | % | 0.00% |
| *40.10* | *Flux braking* | enum | 16 | 0…2 | - | *Disabled* |
| ***44 Maintenance*** | | | | | | |
| *44.01* | *Ontime1 func* | Pb | 16 | 0b00…0b11 | - | 0b01 |
| *44.02* | *Ontime1 src* | Bit pointer | 32 | - | - | *Running* |
| *44.03* | *Ontime1 limit* | UINT32 | 32 | 0…2147483647 | s | 36000000 s |
| *44.04* | *Ontime1 alm sel* | enum | 16 | 0…5 | - | *Mot bearing* |
| *44.05* | *Ontime2 func* | Pb | 16 | 0b00…0b11 | - | 0b01 |
| *44.06* | *Ontime2 src* | Bit pointer | 32 | - | - | *Charged* |
| *44.07* | *Ontime2 limit* | UINT32 | 32 | 0…2147483647 | s | 15768000 s |
| *44.08* | *Ontime2 alm sel* | enum | 16 | 0…5 | - | *Device clean* |
| *44.09* | *Edge count1 func* | Pb | 16 | 0b00…0b11 | - | 0b01 |
| *44.10* | *Edge count1 src* | Bit pointer | 32 | - | - | *Charged* |
| *44.11* | *Edge count1 lim* | UINT32 | 32 | 0…2147483647 | - | 5000 |
| *44.12* | *Edge count1 div* | UINT32 | 32 | 0…2147483647 | - | 1 |
| *44.13* | *Edg cnt1 alm sel* | enum | 16 | 0…5 | - | *Dc-charge* |
| *44.14* | *Edge count2 func* | Pb | 16 | 0b00…0b11 | - | 0b01 |
| *44.15* | *Edge count2 src* | Bit pointer | 32 | - | - | *RO1* |
| *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 | - | 6000 |
| *44.23* | *Val cnt1 alm sel* | enum | 16 | 0…1 | - | *Mot bearing* |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 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 | - | 6000 |
| 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** | | | | | | |
| 45.02 | Energy tariff1 | UINT32 | 32 | 0…21474836.47 | - | 0.65 GBP |
| 45.06 | E tariff unit | enum | 16 | 0…2 | - | Local |
| 45.07 | CO2 Conv factor | REAL | 16 | 0…10 | - | 0.5 |
| 45.08 | Pump ref 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 | SupplyVoltAutoId | enum | 16 | 0…1 | - | Enable |
| 47.04 | Supply voltage | REAL | 16 | 0…1000 | V | 400.0 V |
| **49 Data storage** | | | | | | |
| 49.01 | Data storage1 | UINT32 | 16 | -32768…32767 | - | 0 |
| 49.02 | Data storage2 | UINT32 | 16 | -32768…32767 | - | 0 |
| 49.03 | Data storage3 | UINT32 | 16 | -32768…32767 | - | 0 |
| 49.04 | Data storage4 | UINT32 | 16 | -32768…32767 | - | 0 |
| 49.05 | Data storage5 | UINT32 | 32 | -2147483647 … 2147483647 | - | 0 |
| 49.06 | Data storage6 | UINT32 | 32 | -2147483647 … 2147483647 | - | 0 |
| 49.07 | Data storage7 | UINT32 | 32 | -2147483647 … 2147483647 | - | 0 |
| 49.08 | Data storage8 | UINT32 | 32 | -2147483647 … 2147483647 | - | 0 |
| **50 Fieldbus** | | | | | | |
| 50.01 | FBA enable | enum | 16 | 0…1 | - | Disable |
| 50.02 | Comm loss func | enum | 16 | 0…3 | - | No |
| 50.03 | Comm loss t out | UINT32 | 16 | 0.3…6553.5 | s | 0.3 s |
| 50.04 | FBA ref1 modesel | enum | 16 | 0, 2 | - | Speed |
| 50.05 | FBA ref2 modesel | enum | 16 | 0, 2 | - | Speed |
| 50.06 | FBA act1 tr src | Val pointer | 32 | - | - | P.01.01 |
| 50.07 | FBA act2 tr src | Val pointer | 32 | - | - | P.01.06 |
| 50.08 | FBA sw bit12 src | Bit pointer | 32 | - | - | C.FALSE |
| 50.09 | FBA sw bit13 src | Bit pointer | 32 | - | - | C.FALSE |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| *50.10* | *FBA sw bit14 src* | Bit pointer | 32 | - | - | C.FALSE |
| *50.11* | *FBA sw bit15 src* | Bit pointer | 32 | - | - | C.FALSE |
| *50.15* | *Fb cw used* | Val pointer | 32 | - | - | P.02.22 |
| *50.20* | *Fb main sw func* | Pb | 16 | 0b000…0b111 | - | 0b001 |
| ***51 FBA settings*** | | | | | | |
| *51.01* | *FBA type* | UINT32 | 16 | 0…65535 | - | 0 |
| *51.02* | *FBA par2* | UINT32 | 16 | 0…65535 | - | 0 |
| … | … | … | …. | … | … | … |
| *51.26* | *FBA par26* | UINT32 | 16 | 0…65535 | - | 0 |
| *51.27* | *FBA par refresh* | enum | 16 | 0…1 | - | *Done* |
| *51.28* | *Par table ver* | UINT32 | 16 | - | - | - |
| *51.29* | *Drive type code* | UINT32 | 16 | - | - | - |
| *51.30* | *Mapping file ver* | UINT32 | 16 | - | - | - |
| *51.31* | *D2FBA comm sta* | enum | 16 | 0…6 | - | *Idle* |
| *51.32* | *FBA comm sw ver* | UINT32 | 16 | - | - | - |
| *51.33* | *FBA appl sw ver* | UINT32 | 16 | - | - | - |
| ***52 FBA data in*** | | | | | | |
| *52.01* | *FBA data in1* | UINT32 | 16 | 0…9999 | - | 0 |
| … | … | … | … | … | … | … |
| *52.12* | *FBA data in12* | UINT32 | 16 | 0…9999 | - | 0 |
| ***53 FBA data out*** | | | | | | |
| *53.01* | *FBA data out1* | UINT32 | 16 | 0…9999 | - | 0 |
| … | … | … | … | … | … | … |
| *53.12* | *FBA data out12* | UINT32 | 16 | 0…9999 | - | 0 |
| ***56 Panel display*** | | | | | | |
| *56.01* | *Signal1 param* | UINT32 | | 00.00 … 255.255 | - | 01.03 |
| *56.02* | *Signal2 param* | UINT32 | | 00.00 … 255.255 | - | 01.04 |
| *56.03* | *Signal3 param* | UINT32 | | 00.00 … 255.255 | - | 01.06 |
| *56.04* | *Signal1 mode* | INT32 | | -1…3 | - | *Normal* |
| *56.05* | *Signal2 mode* | INT32 | | -1…3 | - | *Normal* |
| *56.06* | *Signal3 mode* | INT32 | | -1…3 | - | *Normal* |
| *56.07* | *Local ref unit* | UINT32 | | 0…1 | - | *rpm* |
| ***58 Embedded Modbus*** | | | | | | |
| *58.01* | *Protocol ena sel* | UINT32 | 32 | 0…1 | - | *Modbus RTU* |
| *58.03* | *Node address* | UINT32 | 32 | 0…247 | - | 1 |
| *58.04* | *Baud rate* | UINT32 | 32 | 0…6 | - | *9600* |
| *58.05* | *Parity* | UINT32 | 32 | 0…3 | - | *8 none 1* |
| *58.06* | *Control profile* | UINT32 | 32 | 0…3 | - | *ABB Enhanced* |
| *58.07* | *Comm loss t out* | UINT32 | 32 | 0…60000 | ms | 600 |
| *58.08* | *Comm loss mode* | UINT32 | 32 | 0…2 | - | *None* |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 58.09 | Comm loss action | UINT32 | 32 | 0…3 | - | None |
| 58.10 | Refresh settings | UINT32 | 32 | 0…1 | - | Done |
| 58.11 | Reference scale | Pb | 16 | 1…65535 | - | 100 |
| 58.15 | Comm diagnostics | Pb | 16 | 0x0000…0xFFFF | - | 0x0000 |
| 58.16 | Received packets | UINT32 | 32 | 0…65535 | - | 0 |
| 58.17 | Transm packets | UINT32 | 32 | 0…65535 | - | 0 |
| 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 |
| … | … | … | … | … | … | … |
| 58.58 | 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 | % | - |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 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 | % | - |
| 64.22 | AL1 80 to 90% | REAL | 16 | 0…100 | % | - |
| 64.23 | AL1 over 90% | REAL | 16 | 0…100 | % | - |
| 64.24 | AL2 0 to 10% | REAL | 16 | 0…100 | % | - |
| 64.25 | AL2 10 to 20% | REAL | 16 | 0…100 | % | - |
| 64.26 | AL2 20 to 30% | REAL | 16 | 0…100 | % | - |
| 64.27 | AL2 30 to 40% | REAL | 16 | 0…100 | % | - |
| 64.28 | AL2 40 to 50% | REAL | 16 | 0…100 | % | - |
| 64.29 | AL2 50 to 60% | REAL | 16 | 0…100 | % | - |
| 64.30 | AL2 60 to 70% | REAL | 16 | 0…100 | % | - |
| 64.31 | AL2 70 to 80% | REAL | 16 | 0…100 | % | - |
| 64.32 | AL2 80 to 90% | REAL | 16 | 0…100 | % | - |
| 64.33 | AL2 over 90% | REAL | 16 | 0…100 | % | - |
| **75 Pump logic** | | | | | | |
| 75.01 | Operation mode | enum | 16 | 0…3 | - | Off |
| 75.02 | Nbr of pumps | UINT32 | 16 | 0…8 | - | 8 |
| 75.03 | Follower mode | enum | 16 | 0…2 | - | Master speed |
| 75.04 | Follower ref | REAL | 16 | 0…32767 | rpm | 1300 rpm |
| 75.05 | Start speed 1 | UINT32 | 32 | 0…32767 | rpm | 1300 rpm |
| 75.06 | Start speed 2 | UINT32 | 32 | 0…32767 | rpm | 1300 rpm |
| 75.07 | Start speed 3 | UINT32 | 32 | 0…32767 | rpm | 1300 rpm |
| 75.08 | Start speed 4 | UINT32 | 32 | 0…32767 | rpm | 1300 rpm |
| 75.09 | Start speed 5 | UINT32 | 32 | 0…32767 | rpm | 1300 rpm |
| 75.10 | Start speed 6 | UINT32 | 32 | 0…32767 | rpm | 1300 rpm |
| 75.11 | Start speed 7 | UINT32 | 32 | 0…32767 | rpm | 1300 rpm |
| 75.12 | Stop speed 1 | UINT32 | 32 | 0…32767 | rpm | 800 rpm |
| 75.13 | Stop speed 2 | UINT32 | 32 | 0…32767 | rpm | 800 rpm |
| 75.14 | Stop speed 3 | UINT32 | 32 | 0…32767 | rpm | 800 rpm |
| 75.15 | Stop speed 4 | UINT32 | 32 | 0…32767 | rpm | 800 rpm |
| 75.16 | Stop speed 5 | UINT32 | 32 | 0…32767 | rpm | 800 rpm |
| 75.17 | Stop speed 6 | UINT32 | 32 | 0…32767 | rpm | 800 rpm |
| 75.18 | Stop speed 7 | UINT32 | 32 | 0…32767 | rpm | 800 rpm |
| 75.19 | Start delay | UINT32 | 16 | 0…12600 | s | 10 s |
| 75.20 | Stop delay | UINT32 | 16 | 0…12600 | s | 10 s |
| 75.21 | Speed hold on | UINT32 | 16 | 0…100 | s | 0 s |
| 75.22 | Speed hold off | UINT32 | 16 | 0…100 | s | 0 s |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 75.23 | Min pumps allow | UINT32 | 16 | 0…8 | - | 1 |
| 75.24 | Max pumps allow | UINT32 | 16 | 0…8 | - | 8 |
| 75.25 | Drive start dly | UINT32 | 16 | 0…600 | s | 0 s |
| 75.26 | Master speed acc | UINT32 | 32 | 0…1800 | s | 1 s |
| 75.27 | Master speed dec | UINT32 | 32 | 0…1800 | s | 1 s |
| **76 MF communication** | | | | | | |
| 76.01 | Enable MF comm | enum | 16 | 0…1 | - | No |
| 76.02 | Pump node | UINT32 | 16 | 0…8 | - | 1 |
| 76.03 | Master enable | Bit pointer | 32 | - | - | Yes |
| 76.04 | Pump prior sel | Bit pointer | 32 | - | - | Choice 1 |
| 76.05 | Prior choice 1 | UINT32 | 16 | 1…4 | - | 1 |
| 76.06 | Prior choice 2 | UINT32 | 16 | 1…4 | - | 1 |
| 76.07 | Mstr loss action | enum | 16 | 0…1 | - | Const speed |
| 76.08 | Mstr loss delay | UINT32 | 16 | 0…3600 | s | 2 s |
| 76.09 | Start order corr | UINT32 | 16 | 0…1 | - | Optimal |
| 76.10 | Master location | UINT32 | 16 | 0…1 | - | Stable |
| 76.11 | Shared IO enable | UINT32 | 16 | 0…1 | - | No |
| 76.12 | Set as source | Bit pointer | 16 | - | - | No |
| 76.13 | Shared signal 1 | Val pointer | 32 | - | - | AI1 scaled |
| 76.14 | Shared signal 2 | Val pointer | 32 | - | - | AI2 scaled |
| 76.15 | Share lost actn | UINT32 | 16 | 0…3 | - | Alarm |
| 76.16 | Share lost delay | UINT32 | 16 | 0…3600 | s | 10 s |
| **77 Pump sleep** | | | | | | |
| 77.01 | Sleep mode sel | enum | 16 | 0…4 | - | Internal |
| 77.02 | Sleep int sel | Val pointer | 32 | - | - | Speed % |
| 77.03 | Sleep level | REAL | 32 | -32768…32767 | - | 20.00 |
| 77.04 | Sleep delay | UINT32 | 16 | 0…12600 | s | 60 s |
| 77.05 | Sleep ext sel | Bit pointer | 32 | - | - | Not used |
| 77.06 | Sleep boost step | REAL | 16 | 0…32767 | % | 0.00% |
| 77.07 | Sleep boost time | UINT32 | 16 | 0…100 | s | 0 s |
| 77.08 | Wake up mode sel | enum | 16 | 0…3 | - | Wake > ref |
| 77.09 | Wake up ext src | Val pointer | 32 | - | - | Proc act |
| 77.10 | Wake up level | REAL | 32 | -32768…32767 | - | 90.00 |
| 77.11 | Wake up delay | UINT32 | 16 | 0…100 | s | 10 s |
| **78 Pump autochange** | | | | | | |
| 78.01 | Autochg style | enum | 16 | 0…3 | - | No |
| 78.02 | Autochg trad | enum | 16 | 0…1 | - | All |
| 78.03 | Interlock mode | enum | 16 | 0…1 | - | Not used |
| 78.04 | Autochg level | UINT32 | 32 | 0…32767 | rpm | 0 rpm |
| 78.05 | Autochg interval | REAL | 32 | 0…1092.3 | h | 0.00 h |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 78.06 | *Interlock pump 1* | Bit pointer | 32 | - | - | *Not used* |
| 78.07 | *Interlock pump 2* | Bit pointer | 32 | - | - | *Not used* |
| 78.08 | *Interlock pump 3* | Bit pointer | 32 | - | - | *Not used* |
| 78.09 | *Interlock pump 4* | Bit pointer | 32 | - | - | *Not used* |
| 78.10 | *Interlock pump 5* | Bit pointer | 32 | - | - | *Not used* |
| 78.11 | *Interlock pump 6* | Bit pointer | 32 | - | - | *Not used* |
| 78.12 | *Interlock pump 7* | Bit pointer | 32 | - | - | *Not used* |
| 78.13 | *Interlock pump 8* | Bit pointer | 32 | - | - | *Not used* |
| 78.14 | *Runtime change* | enum | 16 | 0…2 | - | *No* |
| 78.15 | *Runtime diff* | UINT32 | 32 | 0…2147483647 | h | 0 h |
| ***79 Level control*** | | | | | | |
| 79.01 | *Level mode* | enum | 16 | 0…2 | - | *Off* |
| 79.02 | *Stopping mode* | enum | 16 | 0…1 | - | *Common stop* |
| 79.03 | *Low level* | REAL | 16 | 0…32767 | % | 5.00% |
| 79.04 | *Low switch* | Bit pointer | 32 | - | - | *Not used* |
| 79.05 | *Stop level* | REAL | 16 | - | - | 10.00% |
| 79.06 | *Start 1 level* | REAL | 16 | - | - | 10.00% |
| 79.07 | *Start 2 level* | REAL | 16 | - | - | 20.00% |
| 79.08 | *Start 3 level* | REAL | 16 | - | - | 30.00% |
| 79.09 | *Start 4 level* | REAL | 16 | - | - | 40.00% |
| 79.10 | *Start 5 level* | REAL | 16 | - | - | 50.00% |
| 79.11 | *Start 6 level* | REAL | 16 | - | - | 60.00% |
| 79.12 | *Start 7 level* | REAL | 16 | - | - | 70.00% |
| 79.13 | *Start 8 level* | REAL | 16 | - | - | 80.00% |
| 79.14 | *High level* | REAL | 16 | - | - | 90.00% |
| 79.15 | *High switch* | Bit pointer | 32 | - | - | *Not used* |
| 79.16 | *Start stop delay* | UINT32 | 16 | 0…3600 | s | 5 s |
| 79.17 | *Random coef* | REAL | 16 | 0…10 | % | 2.0% |
| 79.18 | *Normal speed* | REAL | 16 | 0…32767 | rpm | 1300 rpm |
| 79.19 | *High speed* | REAL | 16 | 0…32767 | rpm | 1500 rpm |
| ***80 Flow calculation*** | | | | | | |
| 80.01 | *Flow calc mode* | enum | 16 | 0…3 | - | *Not used* |
| 80.02 | *Pump inlet sel* | Val pointer | 32 | - | - | *Zero* |
| 80.03 | *Pump outlet sel* | Val pointer | 32 | - | - | *Zero* |
| 80.04 | *HQ curve Q1* | REAL | 16 | 0…32767 | $m^3/h$ | 0.00 $m^3/h$ |
| 80.05 | *HQ curve H1* | REAL | 16 | 0…32767 | m | 0.00 m |
| 80.06 | *HQ curve Q2* | REAL | 16 | 0…32767 | $m^3/h$ | 0.00 $m^3/h$ |
| 80.07 | *HQ curve H2* | REAL | 16 | 0…32767 | m | 0.00 m |
| 80.08 | *HQ curve Q3* | REAL | 16 | 0…32767 | $m^3/h$ | 0.00 $m^3/h$ |
| 80.09 | *HQ curve H3* | REAL | 16 | 0…32767 | m | 0.00 m |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 80.10 | HQ curve Q4 | REAL | 16 | 0…32767 | $m^3$/h | 0.00 $m^3$/h |
| 80.11 | HQ curve H4 | REAL | 16 | 0…32767 | m | 0.00 m |
| 80.12 | HQ curve Q5 | REAL | 16 | 0…32767 | $m^3$/h | 0.00 $m^3$/h |
| 80.13 | HQ curve H5 | REAL | 16 | 0…32767 | m | 0.00 m |
| 80.14 | PQ curve P1 | REAL | 16 | 0…32767 | kW | 0.00 kW |
| 80.15 | PQ curve Q1 | REAL | 16 | 0…32767 | $m^3$/h | 0.00 $m^3$/h |
| 80.16 | PQ curve P2 | REAL | 16 | 0…32767 | kW | 0.00 kW |
| 80.17 | PQ curve Q2 | REAL | 16 | 0…32767 | $m^3$/h | 0.00 $m^3$/h |
| 80.18 | PQ curve P3 | REAL | 16 | 0…32767 | kW | 0.00 kW |
| 80.19 | PQ curve Q3 | REAL | 16 | 0…32767 | $m^3$/h | 0.00 $m^3$/h |
| 80.20 | PQ curve P4 | REAL | 16 | 0…32767 | kW | 0.00 kW |
| 80.21 | PQ curve Q4 | REAL | 16 | 0…32767 | $m^3$/h | 0.00 $m^3$/h |
| 80.22 | PQ curve P5 | REAL | 16 | 0…32767 | kW | 0.00 kW |
| 80.23 | PQ curve Q5 | REAL | 16 | 0…32767 | $m^3$/h | 0.00 $m^3$/h |
| 80.24 | HQ PQ brk point | REAL | 16 | 0…32767 | m | 0.00 m |
| 80.25 | Pump inlet diam | REAL | 16 | 0…32767 | m | 0.00 m |
| 80.26 | Pump outlet diam | REAL | 16 | 0…32767 | m | 0.00 m |
| 80.27 | Sensors hgt diff | REAL | 16 | 0…32767 | m | 0.00 m |
| 80.28 | Pump nom speed | REAL | 16 | 0…32767 | rpm | 1500 rpm |
| 80.29 | Density | REAL | 16 | 0…32767 | kg/$m^3$ | 1000.00 kg/$m^3$ |
| 80.30 | Efficiency | REAL | 16 | 0…100 | % | 100.00% |
| 80.31 | Flow calc gain | REAL | 16 | 0…32767 | - | 1.00 |
| 80.32 | Calc low sp | REAL | 16 | 0…32767 | rpm | 0 rpm |
| 80.33 | Sum flow reset | enum | 16 | 0…1 | - | No |
| **81 Pump protection** | | | | | | |
| 81.01 | Inlet prot ctrl | enum | 16 | 0…3 | - | Not used |
| 81.02 | AI measure inlet | Val pointer | 32 | - | - | AI1 scaled |
| 81.03 | AI in low level | REAL | 16 | 0…32767 | bar | 0.00 bar |
| 81.04 | Very low ctrl | enum | 16 | 0…3 | - | Not sel |
| 81.05 | AI in very low | REAL | 16 | 0…32767 | bar | 0.00 bar |
| 81.06 | DI status inlet | Bit pointer | 32 | - | - | Not used |
| 81.07 | Inlet ctrl dly | UINT32 | 16 | 0…600 | s | 0 s |
| 81.08 | Inlet forced ref | REAL | 16 | 0…32767 | rpm | 0.0 rpm |
| 81.09 | Outlet prot ctrl | enum | 16 | 0…3 | - | Not used |
| 81.10 | AI meas outlet | Val pointer | 32 | - | - | AI1 scaled |
| 81.11 | AI out hi level | REAL | 16 | 0…32767 | bar | 0.00 bar |
| 81.12 | Very high ctrl | enum | 16 | 0…3 | - | Not sel |
| 81.13 | AI out very high | REAL | 16 | 0…32767 | bar | 0.00 bar |
| 81.14 | DI status outlet | Bit pointer | 32 | - | - | Not used |
| 81.15 | Outlet ctr dly | UINT32 | 16 | 0…600 | s | 0 s |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| *81.16* | *Outlet force ref* | REAL | 16 | 0…32767 | rpm | 0.0 rpm |
| *81.17* | *Protect dec time* | UINT32 | 32 | 0…18000 | s | 0 s |
| *81.18* | *Flow source sel* | Val pointer | 32 | - | - | *Flow act* |
| *81.19* | *Flow max prot* | enum | 16 | 0…2 | - | *Not sel* |
| *81.20* | *Flow max level* | REAL | 16 | 0…32767 | $m^3/s$ | 0.00 $m^3/s$ |
| *81.21* | *Flow min prot* | enum | 16 | 0…2 | - | *Not sel* |
| *81.22* | *Flow min level* | REAL | 16 | 0…32767 | $m^3/s$ | 0.00 $m^3/s$ |
| *81.23* | *Flow ctrl delay* | UINT32 | 16 | 0…12600 | s | 0 s |
| *81.24* | *Flow check delay* | UINT32 | 16 | 0…12600 | s | 0 s |
| *81.25* | *Appl prot ctrl* | enum | 16 | 0…2 | - | *Not used* |
| *81.26* | *Prof limit* | REAL | 16 | 0…32767 | % | 0.00% |
| *81.27* | *Prof limit dly* | INT32 | 32 | 0…35791394.1 | h | 0.00 h |
| *81.28* | *Pipefill enable* | Bit pointer | 32 | - | - | *Not used* |
| *81.29* | *Pipefill step* | UINT32 | 16 | 0…32767 | rpm | 50 rpm |
| *81.30* | *Req act change* | REAL | 16 | 0…100 | % | 0.00% |
| *81.31* | *Act change delay* | UINT32 | 16 | 0…100 | s | 3 s |
| *81.32* | *Pid enable dev* | REAL | 16 | 0…100 | % | 10.00% |
| *81.33* | *Pid enb dev dly* | UINT32 | 16 | 0…12600 | s | 1 s |
| *81.34* | *Pipefill timeout* | UINT32 | 16 | 0…12600 | s | 1200 s |
| *81.35* | *Pipefill flt ctr* | enum | 16 | 0…2 | - | *Activate PID* |
| ***82 Pump cleaning*** | | | | | | |
| *82.01* | *Pump clean trig* | Pb | 16 | 0b0000000 … 0b1111111 | - | 0b0100000 |
| *82.02* | *Fwd step* | REAL | 16 | 0…100 | % | 100.0% |
| *82.03* | *Rev step* | REAL | 16 | 0…100 | % | 80.0% |
| *82.04* | *Off time* | UINT32 | 16 | 0…1000 | s | 5 s |
| *82.05* | *Fwd step time* | UINT32 | 16 | 0…1000 | s | 10 s |
| *82.06* | *Rev step time* | UINT32 | 16 | 0…1000 | s | 0 s |
| *82.07* | *Time trig* | INT32 | 32 | 0…35791394.1 | h | 24.00 h |
| *82.08* | *Nbr of steps* | UINT32 | 32 | 0…2147483647 | - | 3 |
| *82.09* | *Supervis source* | Val pointer | 32 | - | - | *Current %* |
| *82.10* | *Supervis limit* | REAL | 16 | 0…32767 | - | 105.0 |
| *82.11* | *Supervis delay* | UINT32 | 16 | 0…600 | s | 10 s |
| *82.12* | *Trig pointer* | Bit pointer | 16 | - | - | *Not used* |
| *82.13* | *Clean max ctrl* | enum | 16 | 0…2 | - | *Alarm* |
| *82.14* | *Clean max number* | UINT32 | 32 | 0…30 | - | 5 |
| *82.15* | *Clean max period* | INT32 | 32 | 0…35791394.1 | h | 1.00 h |
| *82.16* | *Clean step acc* | UINT32 | 32 | 0…32767 | s | 1 s |
| *82.17* | *Clean step dec* | UINT32 | 32 | 0…32767 | s | 1 s |
| ***83 Energy monitoring*** | | | | | | |
| *83.01* | *Energy mon mode* | enum | 16 | 0…3 | - | *Not used* |

| No. | Name | Type | Data len. | Range | Unit | Default (Factory def macro) |
|---|---|---|---|---|---|---|
| 83.02 | *Mon period* | INT32 | 32 | 0…35791394.1 | h | 0.00 h |
| 83.03 | *kWh limit* | UINT32 | 32 | 0…2147483647 | kWh | 0 kWh |
| 83.04 | *Mon tolerance* | UINT32 | 32 | 0…2147483647 | kWh | 0 kWh |
| 83.05 | *Energy mon ctrl* | enum | 16 | 0…1 | - | *Not sel* |
| 83.06 | *Energy reset* | enum | 16 | 0…2 | - | *No* |
| ***94 Ext IO conf*** | | | | | | |
| 94.01 | *Ext IO1 sel* | Val pointer | 32 | 0…4 | - | *None* |
| ***95 Hw configuration*** | | | | | | |
| 95.01 | *Ctrl boardSupply* | enum | 16 | 0…1 | - | *Internal 24V* |
| 95.03 | *Temp inu ambient* | INT32 | 16 | 0…55 | °C | 40 °C |
| ***97 User motor par*** | | | | | | |
| 97.01 | *Use given params* | enum | 16 | 0…1 | - | *NoUserPars* |
| 97.02 | *Rs user* | REAL24 | 32 | 0…0.5 | p.u. | 0.00000 p.u. |
| 97.03 | *Rr user* | REAL24 | 32 | 0…0.5 | p.u. | 0.00000 p.u. |
| 97.04 | *Lm user* | REAL24 | 32 | 0…10 | p.u. | 0.00000 p.u. |
| 97.05 | *SigmaL user* | REAL24 | 32 | 0…1 | p.u. | 0.00000 p.u. |
| 97.09 | *Rs user SI* | REAL24 | 32 | 0…100 | ohm | 0.00000 Ohm |
| 97.10 | *Rr user SI* | REAL24 | 32 | 0…100 | ohm | 0.00000 Ohm |
| 97.11 | *Lm user SI* | REAL24 | 32 | 0…100000 | mH | 0.00 mH |
| 97.12 | *SigL user SI* | REAL24 | 32 | 0…100000 | mH | 0.00 mH |
| ***99 Start-up data*** | | | | | | |
| 99.01 | *Language* | enum | 16 | - | - | *English* |
| 99.05 | *Motor ctrl mode* | enum | 16 | 0…1 | - | *DTC* |
| 99.06 | *Mot nom current* | REAL | 32 | 0…6400 | A | 0.0 A |
| 99.07 | *Mot nom voltage* | REAL | 32 | 1/6 … 2 × $U_N$ | V | 0.0 V |
| 99.08 | *Mot nom freq* | REAL | 32 | 5…500 | Hz | 0.0 Hz |
| 99.09 | *Mot nom speed* | REAL | 32 | 0…30000 | rpm | 0 rpm |
| 99.10 | *Mot nom power* | REAL | 32 | 0…10000 | kW or hp | 0.00 kW |
| 99.11 | *Mot nom cosfii* | REAL24 | 32 | 0…1 | - | 0.00 |
| 99.12 | *Mot nom torque* | INT32 | 32 | 0…2147483.647 | N•m | 0.000 N•m |
| 99.13 | *IDrun mode* | enum | 16 | 0…5 | - | *No* |

8

# Fault tracing

## 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. Selected active faults are shown by *08.20 Pump fault word*.

Alarms can be monitored via alarm words *08.05 Alarm word1* … *08.08 Alarm word4* and *08.21 Pump alarm word*. Alarm information is lost at power switch-off or fault reset.

## Alarm messages generated by the drive

| Code | Alarm (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 2003 | SAFE TORQUE OFF (0xFF7A)<br>*08.05 Alarm word1* b3<br>Programmable alarm: *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 appropriate drive hardware manual, description of parameter *30.07* (page *196*), and *Application guide - Safe torque off function for ACSM1, ACS850 and ACQ810 drives* (3AFE68929814 [English]). |
| 2004 | STO MODE CHANGE (0xFF7A)<br>*08.05 Alarm word1* b4 | 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. |
| 2005 | MOTOR TEMPERATURE (0x4310)<br>*08.05 Alarm word1* b5<br>Programmable alarm: *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.<br>Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc.<br>Check value of alarm limit.<br>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*.<br>Check motor ratings and load.<br>Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc.<br>Check value of alarm limit. |
| 2006 | EMERGENCY OFF (0xF083)<br>*08.05 Alarm word1* b6 | 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)<br>*08.05 Alarm word1* b7 | 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. |

| Code | Alarm (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 2008 | ID-RUN<br>(0xFF84)<br>*08.05 Alarm word1* b8 | 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.<br>Select how motor identification should be performed, parameter *99.13 IDrun mode*.<br>Start identification routines by pressing Start key. |
| 2009 | EMERGENCY STOP<br>(0xF081)<br>*08.05 Alarm word1* b9 | Drive has received emergency stop command (OFF1/OFF3). | Check that it is safe to continue operation.<br>Return emergency stop push button to normal position (or adjust the fieldbus Control Word accordingly).<br>Restart drive. |
| 2013 | DEVICE OVERTEMP<br>(0x4210)<br>*08.05 Alarm word1* b13 | Measured drive temperature has exceeded internal alarm limit. | Check ambient conditions.<br>Check air flow and fan operation.<br>Check heatsink fins for dust pick-up.<br>Check motor power against unit power. |
| 2014 | INTBOARD OVERTEMP<br>(0x7182)<br>*08.05 Alarm word1* b14 | Interface board (between power unit and control unit) temperature has exceeded internal alarm limit. | Let drive cool down.<br>Check for excessive ambient temperature.<br>Check for cooling fan failure.<br>Check for obstructions in the air flow.<br>Check the dimensioning and cooling of the cabinet. |
| 2017 | FIELDBUS COMM<br>(0x7510)<br>*08.06 Alarm word2* b1<br>Programmable alarm: *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 appropriate User's Manual of fieldbus adapter module.<br>Check settings of parameter group *50 Fieldbus*.<br>Check cable connections.<br>Check if communication master is able to communicate. |
| 2018 | LOCAL CTRL LOSS<br>(0x5300)<br>*08.06 Alarm word2* b2<br>Programmable alarm: *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.<br>Check control panel connector.<br>Replace control panel in mounting platform. |
| 2019 | AI SUPERVISION<br>(0x8110)<br>*08.06 Alarm word2* b3<br>Programmable alarm: *13.32 AI superv func* | An analog input has reached limit defined by parameter *13.33 AI superv cw*. | Check analog input source and connections.<br>Check analog input minimum and maximum limit settings. |
| 2020 | FB PAR CONF<br>(0x6320)<br>*08.06 Alarm word2* b4 | The drive does not have a functionality requested by PLC, or requested functionality has not been activated. | Check PLC programming.<br>Check settings of parameter group *50 Fieldbus*. |
| 2021 | NO MOTOR DATA<br>(0x6381)<br>*08.06 Alarm word2* b5 | Parameters in group 99 have not been set. | Check that all the required parameters in group 99 have been set. |

| Code | Alarm (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 2035 | PS COMM<br>(0x5480)<br>*08.07 Alarm word3* b3 | 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<br>(0x6300)<br>*08.07 Alarm word3* b4 | Restoration of backed-up parameters failed. | Contact your local ABB representative. |
| 2037 | CUR MEAS CALIBRATION<br>(0x2280)<br>*08.07 Alarm word3* b5 | Current measurement calibration will occur at next start. | Informative alarm. |
| 2039 | EARTH FAULT<br>(0x2330)<br>*08.07 Alarm word3* b7<br>Programmable alarm: *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.<br>Check for an earth fault in motor or motor cables by measuring the insulation resistances of motor and motor cable.<br>If no earth fault can be detected, contact your local ABB representative. |
| 2040 | AUTORESET<br>(0x6080)<br>*08.07 Alarm word3* b8 | A fault is to be autoreset. | Informative alarm. See parameter group *32 Automatic reset*. |
| 2041 | MOTOR NOM VALUE<br>(0x6383)<br>*08.07 Alarm word3* b9 | 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. |
| 2043 | STALL<br>(0x7121)<br>*08.07 Alarm word3* b11<br>Programmable alarm: *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.<br>Check fault function parameters. |
| 2044 | LCURVE<br>(0x2312)<br>*08.07 Alarm word3* b12<br>Programmable alarm: *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<br>(0x6320)<br>*08.07 Alarm word3* b13 | 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<br>(0x6320)<br>*08.07 Alarm word3* b14 | The *U/f* (voltage/frequency) curve has been incorrectly or inconsistently defined. | Check the settings of the parameters in group *38 Flux ref*. |
| 2048 | OPTION COMM LOSS<br>(0x7000)<br>*08.08 Alarm word4* b0 | Communication between drive and option module (FEN-xx and/or FIO-xx) is lost. | Check that option modules are properly connected to Slot 1 (or) Slot 2.<br>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), other information | Cause | What to do |
|---|---|---|---|
| 2049 | MOTTEMPAL2<br>(0x4313)<br>*08.08 Alarm word4* b2<br>Programmable alarm: *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.<br>Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc.<br>Check value of alarm limit.<br>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*.<br>Check motor ratings and load.<br>Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc.<br>Check value of alarm limit. |
| 2050 | IGBTOLALARM<br>(0x5482)<br>*08.08 Alarm word4* b3 | 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. |
| 2051 | IGBTTEMPALARM<br>(0x4210)<br>*08.08 Alarm word4* b4 | Drive IGBT temperature is excessive. | Check ambient conditions.<br>Check air flow and fan operation.<br>Check heatsink fins for dust pick-up.<br>Check motor power against drive power. |
| 2052 | COOLALARM<br>(0x4290)<br>*08.08 Alarm word4* b5 | 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 appropriate *Hardware Manual.*<br>Check drive module cooling air flow and fan operation.<br>Check inside of cabinet and heatsink of drive module for dust pick-up. Clean whenever necessary. |
| 2053 | MENU CHG PASSWORD REQ<br>(0x6F81) | Loading a parameter listing requires a password. | Enter password at parameter *16.03 Pass code*. |
| 2054 | MENU CHANGED<br>(0x6F82)<br>*08.08 Alarm word4* b6 | A different parameter listing is being loaded. | Informative alarm. |

| Code | Alarm (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 2055 | DEVICE CLEAN (0x5080) | Maintenance counter alarm. | See parameter group *44 Maintenance*.<br>**Note:** Any maintenance counter alarm sets bit 8 of *08.08 Alarm word4*. |
| 2056 | COOLING FAN (0x5081) | | |
| 2057 | ADD COOLING (0x5082) | | |
| 2058 | CABINET FAN (0x5083) | | |
| 2059 | DC CAPACITOR (0x5084) | | |
| 2060 | MOTOR BEARING (0x738C) | | |
| 2061 | MAIN CONTACTOR (0x548D) | | |
| 2062 | RELAY OUTPUT SW (0x548E) | | |
| 2063 | MOTOR START COUNT (0x6180) | | |
| 2064 | POWER UP COUNT (0x6181) | | |
| 2065 | DC CHARGE COUNT (0x6182) | | |
| 2066 | ONTIME1 ALARM (0x5280) | | |
| 2067 | ONTIME2 ALARM (0x5281) | | |
| 2068 | EDGE1 ALARM (0x5282) | | |
| 2069 | EDGE2 ALARM (0x5283) | | |
| 2070 | VALUE1 ALARM (0x5284) | | |
| 2071 | VALUE2 ALARM (0x5285) | | |
| 2072 | DC NOT CHARGED (0x3250)<br>*08.08 Alarm word4* b9 | 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)<br>*08.08 Alarm word4* b10 | Speed controller autotune routine did not finish successfully. | See parameter *23.20 PI tune mode*. |
| 2074 | START INTERLOCK (0xF082)<br>*08.08 Alarm word4* b11 | No Start interlock signal received. | Check circuit connected to DIIL input. |
| 2076 | TEMP MEAS FAILURE (0x4211)<br>*08.08 Alarm word4* b7 | Problem with internal temperature measurement of the drive. | Contact your local ABB representative. |

| Code | Alarm (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 2077 | EFB COMM LOSS<br>(0x7540)<br>*08.08 Alarm word4* b12 | Embedded fieldbus interface has been taken into use, and there is a communication break between the drive and the master station. | Check:<br>• selection of the parameter which enables/disables EFB communication (*58.01 Protocol ena sel*)<br>• EFB connection at terminal XD2D on the JCON board<br>• status of the fieldbus master (online/ offline)<br>• settings of the communication supervision function (parameter *58.09 Comm loss action*). |
| 2201 | PIPEFILL TIMEOUT<br>*08.09 Alarm word5* b0<br>*08.21 Pump alarm word* b9 | Maximum allowed time for the Pipefill function exceeded. | Check the pump system.<br>Check parameters *81.28*….*81.35*. |
| 2202 | MIN FLOW<br>*08.09 Alarm word5* b1<br>*08.21 Pump alarm word* b0 | Measured flow below minimum limit. | Check the pump system for reasons such as leaks that might cause a loss of measured flow.<br>Check parameters *81.18*….*81.24*. |
| 2203 | MAX FLOW<br>*08.09 Alarm word5* b2<br>*08.21 Pump alarm word* b1 | Measured flow above maximum limit. | Check the pump system for reasons that might cause an increase in measured flow.<br>Check parameters *81.18*….*81.24*. |
| 2204 | LOW PRESSURE<br>*08.09 Alarm word5* b3<br>*08.21 Pump alarm word* b2 | Pressure at pump inlet too low. | Check for a closed valve on the inlet side of the pump.<br>Check piping for leaks. |
| 2205 | HIGH PRESSURE<br>*08.09 Alarm word5* b4<br>*08.21 Pump alarm word* b3 | Pressure at pump outlet too high. | Check piping for blocks. |
| 2206 | VERY LOW PRESS<br>*08.09 Alarm word5* b5<br>*08.21 Pump alarm word* b4 | Pressure at pump inlet too low. | Check for a closed valve on the inlet side of the pump.<br>Check piping for leaks. |
| 2207 | VERY HIGH PRESS<br>*08.09 Alarm word5* b6<br>*08.21 Pump alarm word* b5 | Pressure at pump outlet too high. | Check piping for blocks. |
| 2208 | PROFILE HIGH<br>*08.09 Alarm word5* b7<br>*08.21 Pump alarm word* b6 | Application profile protection limit exceeded (see parameters *81.25*…*81.27*). | Check the piping for leaks.<br>Check the general condition of the components of the pumping station. |
| 2209 | MAX CLEANINGS<br>*08.09 Alarm word5* b8<br>*08.21 Pump alarm word* b7 | Maximum number of cleaning sequences exceeded (see parameter group *82 Pump cleaning*). | Check for reasons that might have had an increasing effect on the monitored signal (parameter *82.09*). For example, increased viscosity of the fluid, or faulty pump bearings may increase the current drawn by the motor, and trigger the cleaning sequence more frequently. |
| 2210 | ALL PUMPS INLOCKD<br>*08.09 Alarm word5* b9<br>*08.21 Pump alarm word* b10 | All interlock signals are off, indicating no pumps are available. | Check the interlock settings in parameter group *78 Pump autochange*.<br>Check that the pumps are switched on.<br>Check the interlock wiring from the pumps. |
| 2211 | ENERGY LIMIT<br>*08.09 Alarm word5* b10<br>*08.21 Pump alarm word* b11 | Energy consumption limit exceeded (see parameter group *83 Energy monitoring*). | Check for reasons for increased energy consumption. |

| Code | Alarm (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 2212 | DATE WRONG<br>*08.09 Alarm word5* b11<br>*08.21 Pump alarm word* b12 | Date has not been set. | Set the date and time (page *37*). |
| 2215 | BOOSTING<br>*08.09 Alarm word5* b14<br>*08.21 Pump alarm word* b15 | Sleep boost is active. | Informative alarm. |
| 2216 | PIPE FILLING<br>*08.09 Alarm word5* b15<br>*08.21 Pump alarm word* b16 | The soft pipefill function is being performed. | Informative alarm. |
| 2217 | NO MORE PUMPS<br>*08.10 Alarm word6* b0<br>*08.21 Pump alarm word* b17 | No further pumps are available for starting. | Check that all appropriate pumps are switched on. |
| 2218 | CLEANING<br>*08.10 Alarm word6* b1<br>*08.21 Pump alarm word* b8 | A pump cleaning sequence is in progress. | Informative alarm. |
| 2219 | AUTOCHANGE<br>*08.10 Alarm word6* b2<br>*08.21 Pump alarm word* b18 | The Autochange function is being performed. | Informative alarm. |
| 2220 | SLEEPING<br>*08.10 Alarm word6* b3<br>*08.21 Pump alarm word* b19 | The drive has entered sleep mode. | Informative alarm. |
| 2221 | START DELAY<br>*08.10 Alarm word6* b4<br>*08.21 Pump alarm word* b20 | A pump will start after the start delay has elapsed. | Informative alarm. |
| 2222 | LC TANK FULL<br>*08.10 Alarm word6* b5<br>*08.21 Pump alarm word* b23 | The level of the liquid in the container is very high (the source selected by parameter *79.15 High switch* is 1). | Informative alarm. |
| 2223 | LC TANK EMPTY<br>*08.10 Alarm word6* b6<br>*08.21 Pump alarm word* b24 | The level of the liquid in the container is very low (the source selected by parameter *79.04 Low switch* is 1). | Informative alarm. |
| 2224 | MF MASTER LOST<br>*08.10 Alarm word6* b7<br>*08.21 Pump alarm word* b21 | The drive cannot detect a master on the drive-to-drive link, and is not itself allowed to become master. | Check that there are drives on the drive-to-drive link that are allowed to become master.<br>Check the wiring of the drive-to-drive link. |
| 2225 | MF NO SHARED DATA<br>*08.10 Alarm word6* b8<br>*08.21 Pump alarm word* b25 | Shared signals not received. | Check that at least one drive has signal sharing enabled (parameter *76.12 Set as source*).<br>Check the status, communication settings and wiring of the drive that is sharing its signals. |
| 2400 | SOLUTION ALARM<br>(0x6F80)<br>*08.08 Alarm word4* b1 | Alarm generated by custom application program. | Check custom application program. |

## Fault messages generated by the drive

| Code | Fault (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 0001 | OVERCURRENT<br>(0x2310) | Output current has exceeded internal fault limit. | Check motor load.<br>Check acceleration times in parameter group *22 Speed ref ramp*.<br>Check motor and motor cable (including phasing and delta/star connection).<br>Check that the start-up data in parameter group 99 corresponds to the motor rating plate.<br>Check that there are no power factor correction capacitors or surge absorbers in motor cable. |
| 0002 | DC OVERVOLTAGE<br>(0x3210) | Excessive intermediate circuit DC voltage | Check that overvoltage controller is on, parameter *47.01 Overvolt ctrl*.<br>Check mains for static or transient overvoltage.<br>Check deceleration time.<br>Use coast-to-stop function (if applicable). |
| 0003 | DEVICE OVERTEMP<br>(0x4210) | Measured drive temperature has exceeded internal fault limit. | Check ambient conditions.<br>Check air flow and fan operation.<br>Check heatsink fins for dust pick-up.<br>Check motor power against unit power. |
| 0004 | SHORT CIRCUIT<br>(0x2340) | Short-circuit in motor cable(s) or motor | Check motor and motor cable.<br>Check there are no power factor correction capacitors or surge absorbers in motor cable. |
| 0005 | DC UNDERVOLTAGE<br>(0x3220) | Intermediate circuit DC voltage is not sufficient due to missing mains phase, blown fuse or rectifier bridge internal fault. | Check mains supply and fuses. |
| 0006 | EARTH FAULT<br>(0x2330)<br>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.<br>Check that there is no earth fault in motor or motor cables:<br>- measure insulation resistances of motor and motor cable.<br>If no earth fault can be detected, contact your local ABB representative. |
| 0007 | FAN FAULT<br>(0xFF83) | Fan is not able to rotate freely or fan is disconnected. Fan operation is monitored by measuring fan current. | Check fan operation and connection. |
| 0013 | CURR MEAS GAIN<br>(0x3183) | Difference between output phase U2 and W2 current measurement gain is too great. | Contact your local ABB representative. |
| 0014 | CABLE CROSS CON<br>(0x3181)<br>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. |

| Code | Fault (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 0015 | SUPPLY PHASE (0x3130)<br>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.<br>Check for input power supply imbalance. |
| 0016 | MOTOR PHASE (0x3182)<br>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*.<br>Check that the drive is dimensioned correctly according to the motor. |
| | 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<br>• *20.01 Maximum speed* > (0.55 × *99.09 Mot nom speed*) > (0.50 × synchronous speed),<br>• *20.02 Minimum speed* ≤ 0, and<br>• 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: 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. |

| Code | Fault (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 0019 | CURR V2 MEAS<br>(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<br>(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. |
| 0021 | STO1 LOST<br>(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 appropriate drive hardware manual, description of parameter *30.07* (page *196*), and *Application guide - Safe torque off function for ACSM1, ACS850 and ACQ810 drives* (3AFE68929814 [English]). |
| 0022 | STO2 LOST<br>(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<br>(0xFF7A) | Error in changing Safe torque off supervision, i.e. parameter *30.07 Sto diagnostic* setting could not be changed to value *Fault*. | Contact your local ABB representative. |
| 0024 | INTBOARD OVERTEMP<br>(0x7182) | Interface board (between power unit and control unit) temperature has exceeded internal fault limit. | Let drive cool down.<br>Check for excessive ambient temperature.<br>Check for cooling fan failure.<br>Check for obstructions in the air flow.<br>Check the dimensioning and cooling of the cabinet. |
| 0027 | PU LOST<br>(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<br>(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<br>(0x9000) | Fault in external device. (This information is configured through one of programmable digital inputs.) | Check external devices for faults.<br>Check parameter *30.01 External fault* setting. |
| 0031 | SAFE TORQUE OFF<br>(0xFF7A)<br>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 *Fault*. | Check safety circuit connections. For more information, see appropriate drive hardware manual, and *Application guide - Safe torque off function for ACSM1, ACS850 and ACQ810 drives* (3AFE68929814 [English]). |

| Code | Fault (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 0032 | OVERSPEED<br>(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*.<br>Check adequacy of motor braking torque.<br>Check applicability of torque control. |
| 0036 | LOCAL CTRL LOSS<br>(0x5300)<br>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.<br>Check control panel connector.<br>Replace control panel in mounting platform. |
| 0037 | NVMEM CORRUPTED<br>(0x6320) | Drive internal fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0038 | OPTIONCOMM LOSS<br>(0x7000) | Communication between drive and option module (FIO-xx) is lost. | Check that option modules are properly connected to Slot 1 and (or) Slot 2.<br>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. |
| 0045 | FIELDBUS COMM<br>(0x7510)<br>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 appropriate User's Manual of fieldbus adapter module.<br>Check settings of parameter group *50 Fieldbus*.<br>Check cable connections.<br>Check if communication master is able to communicate. |
| 0046 | FB MAPPING FILE<br>(0x6306) | Drive internal fault | Contact your local ABB representative. |
| 0047 | MOTOR OVERTEMP<br>(0x4310)<br>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*. | Check motor ratings and load.<br>Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc.<br>Check value of alarm limit.<br>Check motor thermal model settings (parameters *31.09*…*31.14*). |
| | | Measured motor temperature has exceeded fault limit defined by parameter *31.04 Mot temp1 fltLim*. | Check that actual number of sensors corresponds to value set by parameter *31.02 Mot temp1 src*.<br>Check motor ratings and load.<br>Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc.<br>Check value of alarm limit. |
| 0049 | AI SUPERVISION<br>(0x8110)<br>Programmable fault: *13.32 AI superv func* | An analog input has reached limit defined by parameter *13.33 AI superv cw*. | Check analog input source and connections.<br>Check analog input minimum and maximum limit settings. |
| 0055 | TECH LIB<br>(0x6382) | Resettable fault generated by a technology library. | Refer to the documentation of the technology library. |

| Code | Fault (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 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.<br>Check settings of parameter group *50 Fieldbus*. |
| 0059 | STALL (0x7121)<br>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.<br>Check fault function parameters. |
| 0060 | LOAD CURVE (0x2312)<br>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*. |
| 0063 | MOTOR TEMP2 (0x4313)<br>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.<br>Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc.<br>Check value of alarm limit.<br>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*.<br>Check motor ratings and load.<br>Let motor cool down. Ensure proper motor cooling: Check cooling fan, clean cooling surfaces, etc.<br>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.<br>Check air flow and fan operation.<br>Check heatsink fins for dust pick-up.<br>Check motor power against drive power. |
| 0066 | COOLING (0x4290) | Drive module temperature is excessive. | Check setting of parameter *95.03 Temp inu ambient*.<br>Check ambient temperature. If it exceeds 40 °C (104 °F), ensure that load current does not exceed derated load capacity of drive. See appropriate *Hardware Manual.*<br>Check drive module cooling air flow and fan operation.<br>Check inside of cabinet and heatsink of drive module for dust pick-up. Clean whenever necessary. |

| Code | Fault (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 0070 | TEMP MEAS FAILURE (0x4211) | Problem with internal temperature measurement of the drive. | Contact your local ABB representative. |
| 0071 | EFB COMM LOSS (0x7540) | Embedded fieldbus interface has been taken into use, and there is a communication break between the drive and the master station. | Check:<br>• selection of the parameter which enables/disables EFB communication (*58.01 Protocol ena sel*)<br>• EFB connection at terminal XD2D on the JCON board<br>• status of the fieldbus master (online/ offline)<br>• settings of the communication supervision function (parameter *58.09 Comm loss action*). |
| 0201 | T2 OVERLOAD (0x0201) | Firmware time level 2 overload<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0202 | T3 OVERLOAD (0x6100) | Firmware time level 3 overload<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0203 | T4 OVERLOAD (0x6100) | Firmware time level 4 overload<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0204 | T5 OVERLOAD (0x6100) | Firmware time level 5 overload<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0205 | A1 OVERLOAD (0x6100) | Application time level 1 fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0206 | A2 OVERLOAD (0x6100) | Application time level 2 fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0207 | A1 INIT FAULT (0x6100) | Application task creation fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0208 | A2 INIT FAULT (0x6100) | Application task creation fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0209 | STACK ERROR (0x6100) | Drive internal fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0210 | FPGA ERROR (0xFF61) | Drive internal fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0301 | UFF FILE READ (0x6300) | File read error<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |

| Code | Fault (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 0302 | APPL DIR CREATION (0x6100) | Drive internal fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0303 | FPGA CONFIG DIR (0x6100) | Drive internal fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0304 | PU RATING ID (0x5483) | Drive internal fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0305 | RATING DATABASE (0x6100) | Drive internal fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0306 | LICENSING (0x6100) | Drive internal fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0307 | DEFAULT FILE (0x6100) | Drive internal fault<br>**Note:** This fault cannot be reset. | Contact your local ABB representative. |
| 0308 | APPLFILE PAR (0x6300) | Corrupted application file<br>**Note:** This fault cannot be reset. | Reload application.<br>If fault is still active, contact your local ABB representative. |
| 0309 | APPL LOADING (0x6300) | Corrupted application file<br>**Note:** This fault cannot be reset. | Reload application.<br>If fault is still active, contact your local ABB representative. |
| 0310 | USERSET LOAD (0xFF69) | Loading of user set is not successfully completed because:<br>- requested user set does not exist<br>- user set is not compatible with drive program<br>- drive has been switched off during loading. | Reload. |
| 0311 | USERSET SAVE (0xFF69) | User set is not saved because of memory corruption. | Check the setting of parameter *95.01 Ctrl boardSupply*.<br>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<br>**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. |

| Code | Fault (fieldbus code), other information | Cause | What to do |
|---|---|---|---|
| 0318 | MENU HIDING | Menu hiding file missing or corrupted. | Reload application.<br>Contact your local ABB representative. |
| 0401 | PIPEFILL TOUT<br>*08.20 Pump fault word* b7 | Maximum allowed time for the Pipefill function exceeded. | Check the pump system.<br>Check parameters *81.28*….*81.35*. |
| 0402 | MIN FLOW<br>*08.20 Pump fault word* b0 | Measured flow below minimum limit. | Check the pump system for reasons such as leaks that might cause a loss of measured flow.<br>Check parameters *81.18*….*81.24*. |
| 0403 | MAX FLOW<br>*08.20 Pump fault word* b1 | Measured flow above maximum limit. | Check the pump system for reasons that might cause an increase in measured flow.<br>Check parameters *81.18*….*81.24*. |
| 0404 | LOW PRESSURE<br>*08.20 Pump fault word* b2 | Pressure at pump inlet too low. | Check for a closed valve on the inlet side of the pump.<br>Check piping for leaks. |
| 0405 | HIGH PRESSURE<br>*08.20 Pump fault word* b3 | Pressure at pump outlet too high. | Check piping for blocks. |
| 0406 | VERY LOW PRESS<br>*08.20 Pump fault word* b4 | Pressure at pump inlet too low. | Check for a closed valve on the inlet side of the pump.<br>Check piping for leaks. |
| 0407 | VERY HIGH PRESS<br>*08.20 Pump fault word* b5 | Pressure at pump outlet too high. | Check piping for blocks. |
| 0408 | MAX CLEANINGS<br>*08.20 Pump fault word* b6 | Maximum number of cleaning sequences exceeded (see parameter group *82 Pump cleaning*). | Check for reasons that might have had an increasing effect on the monitored signal (parameter *82.09*). For example, increased viscosity of the fluid, or faulty pump bearings may increase the current drawn by the motor, and trigger the cleaning sequence more frequently. |
| 0409 | MF MASTER LOST<br>*08.20 Pump fault word* b8 | The drive cannot detect a master on the drive-to-drive link, and is not itself allowed to become master. | Check that there are drives on the drive-to-drive link that are allowed to become master.<br>Check the wiring of the drive-to-drive link. |
| 0410 | MF NO SHARED DATA<br>*08.20 Pump fault word* b9 | Shared signals not received. | Check that at least one drive has signal sharing enabled (parameter *76.12*).<br>Check the status, communication settings and wiring of the drive that is sharing its signals. |

9

# Control through the embedded fieldbus interface

## What this chapter contains

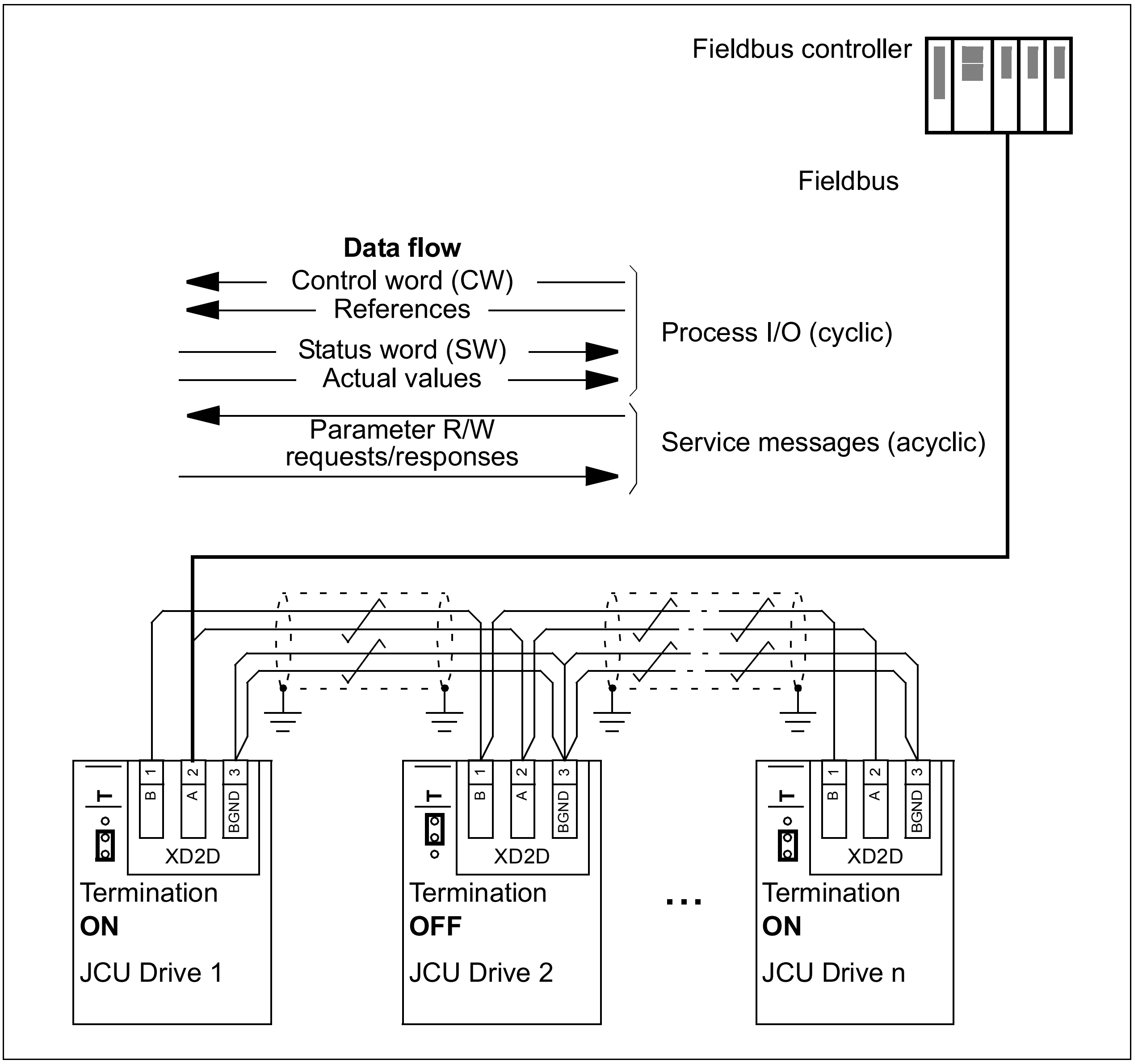
The chapter describes how the drive can be controlled by external devices over a communication network (fieldbus) using the embedded fieldbus interface.

## System overview

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

Connect the fieldbus 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.

## 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.

The new settings will take effect when the drive is powered up the next time, or when parameter *58.10 Refresh settings* is activated.

| Parameter | Setting for fieldbus control | Function/Information |
|---|---|---|
| COMMUNICATION INITIALIZATION | | |
| *50.15 Fb cw used* | P.02.36 (default) | Selects the address of the Fieldbus control word in use (*02.36 EFB main cw*). |
| *58.01 Protocol ena sel* | *Modbus RTU* | Initializes embedded fieldbus communication. |
| EMBEDDED MODBUS CONFIGURATION | | |
| *58.03 Node address* | 1 (default) | Node address. There may not be two nodes with the same node address online. |
| *58.04 Baud rate* | 9600 (default) | Defines the communication speed of the link. Use the same setting as in the master station. |
| *58.05 Parity* | *8 none 1* (default) | Selects the parity and stop bit setting. Use the same setting as in the master station. |
| *58.06 Control profile* | *ABB Enhanced* (default) | Selects the communication profile used by the drive. See section *Basics of the embedded fieldbus interface* on page *336*. |
| *58.07 Comm loss t out* | 600 (default) | Defines the timeout limit for the EFB communication monitoring. |
| *58.08 Comm loss mode* | *None* (default) | Enables/disables EFB communication loss monitoring and defines the means for resetting the counter of the communication loss delay. |
| *58.09 Comm loss action* | *None* (default) | Defines the drive operation after the EFB communication loss monitoring awakes. |
| *58.10 Refresh settings* | *Done* (default) | Refreshes the settings of parameters *58.01…58.09*. |
| *58.30 Transmit delay* | 0 (default) | Defines the delay time which the slave waits until it sends a response. |
| *58.31 Ret app errors* | *Yes* (default) | Selects whether the drive returns Modbus exception codes or not. |
| *58.32 Word order* | *LSW MSW* (default) | Defines the order of the data words in the Modbus frame. |

| Parameter | Setting for fieldbus control | Function/Information |
|---|---|---|
| *58.35 Data I/O 1*<br>… …<br>*58.58 Data I/O 24* | 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. |

## 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* | *FBA* | 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* | *FBA* | 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 *FBA* 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 the Embedded fieldbus reference REF1, set parameter *21.01* to *EFB ref1* and keep parameter *12.01* to its default value (C.FALSE). | | |
| REFERENCE SCALING | | |
| *50.04 FBA ref1 modesel* | *Raw data*<br>*Speed* | Defines the fieldbus reference REF1 scaling. Selects also the fieldbus actual signal act1 when set to *Speed*. |
| *50.05 FBA ref2 modesel* | *Raw data*<br>*Speed* | Defines the fieldbus reference REF2 scaling. Selects also the fieldbus actual signal act2 when set to *Speed*. |
| ACTUAL VALUE ACT1 AND ACT 2 SELECTION (if *50.04* or *50.05* has value *Raw data*). | | |
| *50.06 FBA act1 tr src* | Any | Selects the source for fieldbus actual value act1 when parameter *50.04 FBA ref1 modesel* is set to *Raw data*. |

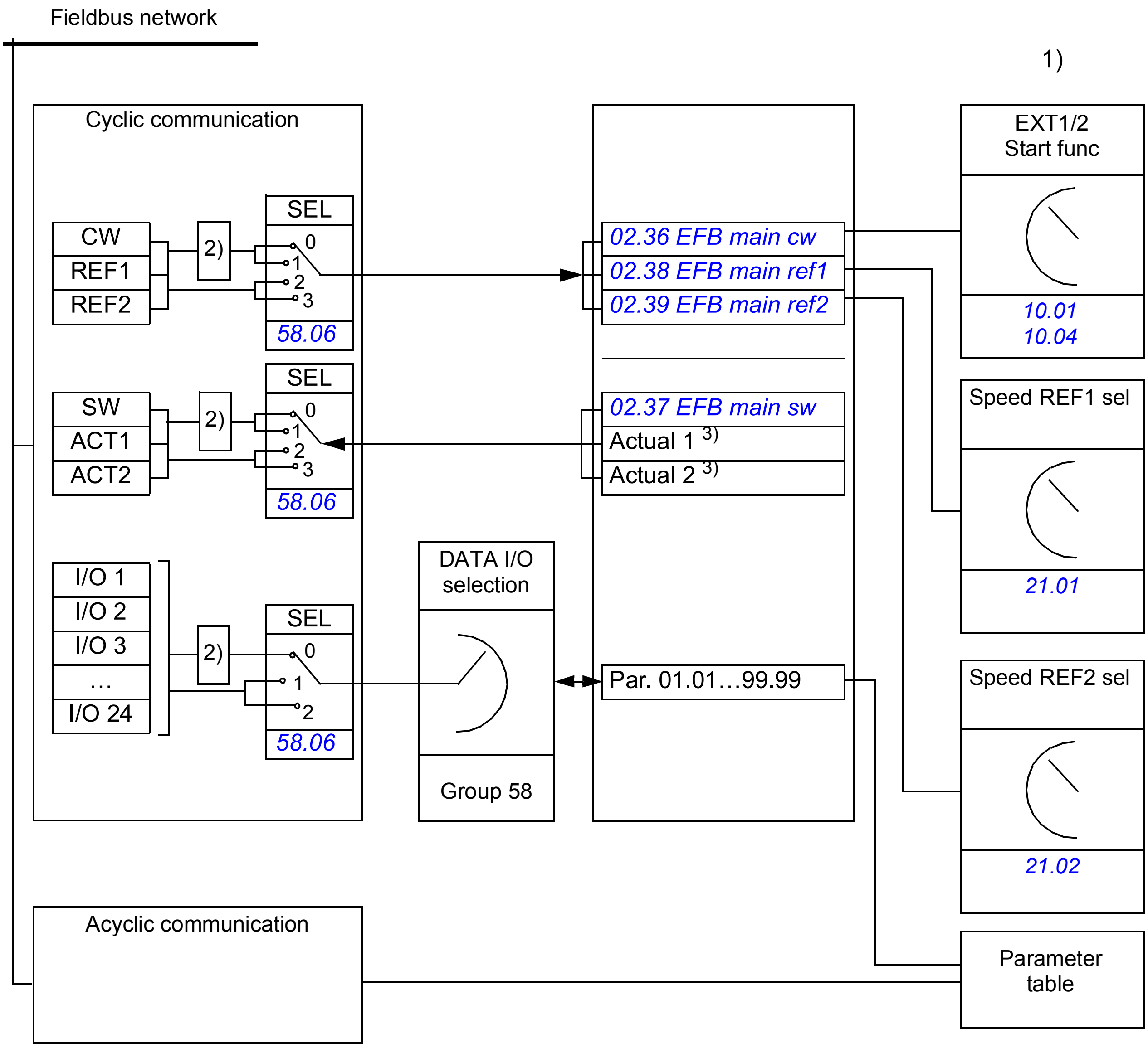
| **Parameter** | **Setting for fieldbus control** | **Function/Information** |
|---|---|---|
| *50.07 FBA act2 tr src* | Any | Selects the source for fieldbus actual value act2 when parameter *50.05 FBA ref2 modesel* is set to *Raw data*. |

| SYSTEM CONTROL INPUTS | | |
|---|---|---|
| *16.07 Param save* | *Save* (restores to *Done*) | 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 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 (0) *ABB Classic* or (1) *ABB Enhanced*. See section *About the EFB communication profiles* on page *338*.
3) See parameter *50.04 FBA ref1 modesel* and *50.05 FBA 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 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. In the embedded fieldbus communication, the CW is written to drive parameter *02.36 EFB main cw* from where it can be used in the control of the drive. The Fieldbus CW is either written to the drive Control word as it is, or the data is converted. See section *About the EFB communication profiles* on page *338*.

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 status word is either written to the fieldbus SW as it is or the data is converted. See section *About the EFB communication profiles* on page *338*.

## 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, 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 *338*.

## 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 *338*.

## Data inputs/outputs

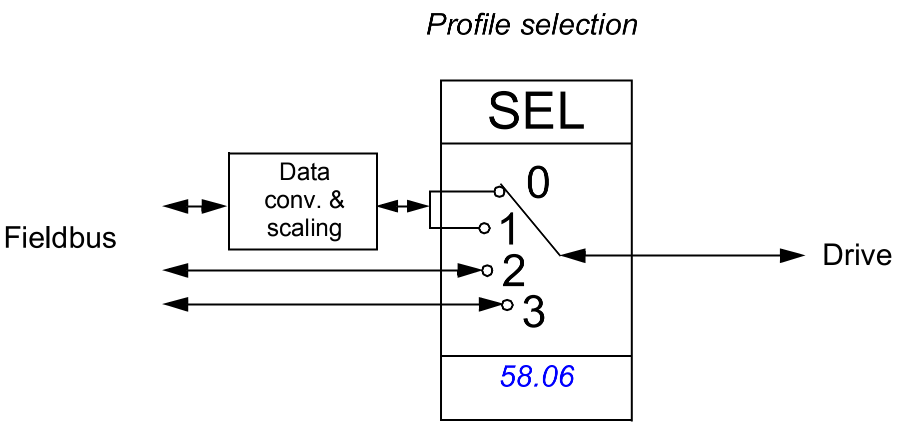
Data input/output (I/O) are 16-bit or 32-bit words containing selected drive parameter values. Parameters *58.35 Data I/O 1* … *58.58 Data I/O 24* define the addresses from which the master either reads data (input) or to which it writes data (output).

## About the EFB communication profiles

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

- if packed boolean words are converted and how
- if the signal values are scaled and how
- how the 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, 16-bit DCU profile or 32-bit DCU 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* are:

- (0) *ABB Classic*
- (1) *ABB Enhanced*
- (2) *DCU 16-bit*
- (2) *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 *343*.

| 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.<br>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**.<br>**Warning:** Ensure motor and driven machine can be stopped using this stop mode. |
| 3 | INHIBIT_ OPERATION | 1 | Proceed to **OPERATION ENABLED**.<br>**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.<br>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**.<br>**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. |

| Bit | Name | Value | STATE/Description |
|---|---|---|---|
| 7 | RESET | 0=>1 | Fault reset if an active fault exists. Proceed to **SWITCH-ON INHIBITED**.<br>**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.<br>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 control location is parameterized to be selected from fieldbus. |
| | | 0 | Select External Control Location EXT1. Effective if control location is parameterized to be selected from fieldbus. |
| 12 …15 | Reserved | | |

## Status word for the ABB Drives profiles

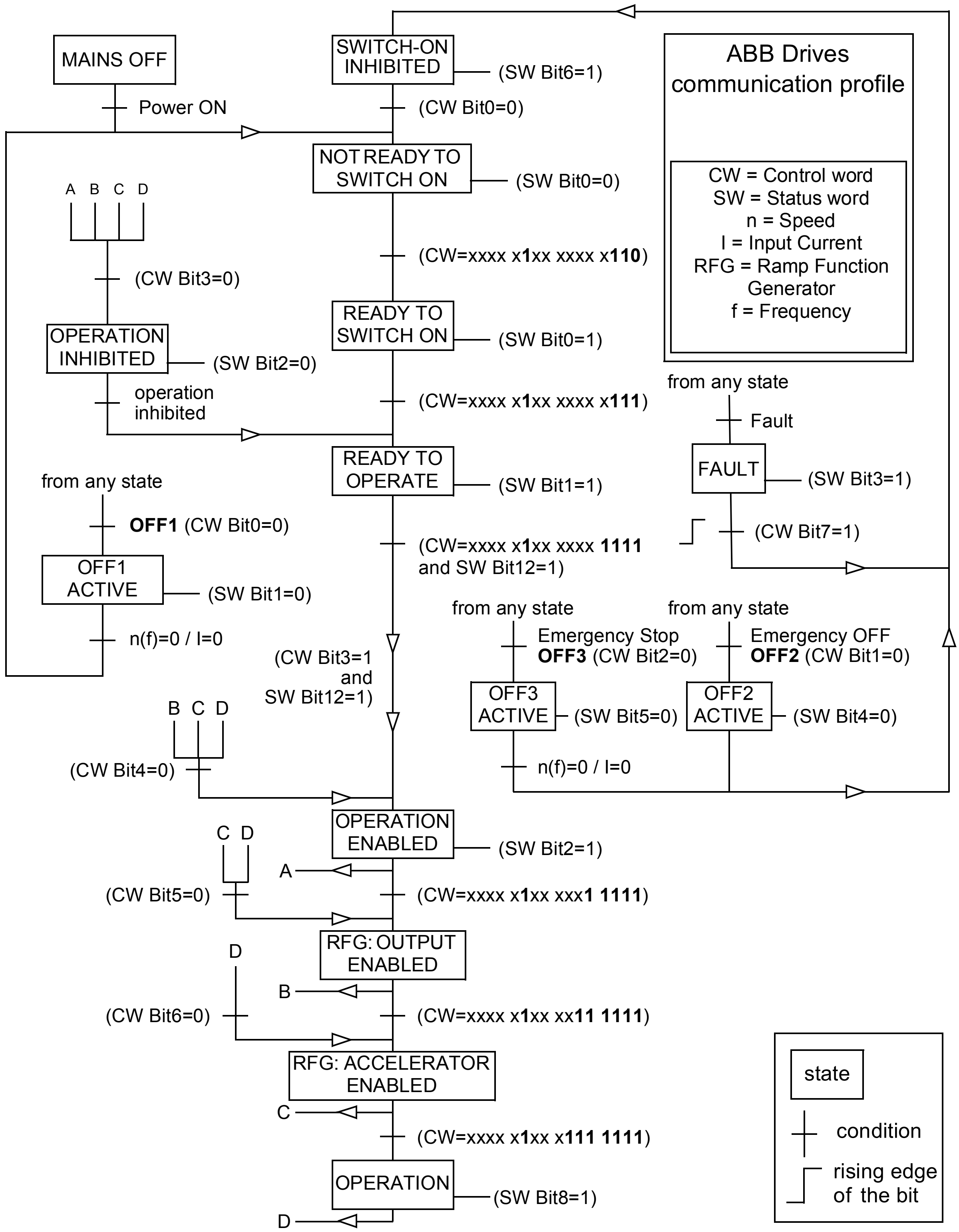
The table below shows the Fieldbus status word for both ABB Drives profiles. The embedded fieldbus interface converts the Drive status word (*02.37 EFB main sw*) to this form for the transfer in the fieldbus. The upper case boldface text refers to the states shown in *State transition diagram for the ABB Drives profiles* on page 343.

| 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 | | |

| 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 embedded 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 *339* and *Status word for the ABB Drives profiles* on page *341*.



## 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 FBA ref1 modesel* and *50.05 FBA 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 *Raw data*, the fieldbus reference is scaled as shown in the table below.

| Fieldbus reference REF1 or REF2 [integer] | Corresponding reference in the drive [rpm] |
|---|---|
| 32 767 | k × 0.5<br>(k = value of parameter *58.11*) |
| 0 | 0 |
| -32 768 | k × -0.5<br>(k = value of parameter *58.11*) |

## 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 written into fieldbus actual values, ACT1 and ACT2. Parameters *50.04 FBA ref1 modesel* and *50.05 FBA ref2 modesel* both select the drive actual signals and define the scaling as follows:

- If you select value *Speed*, the drive actual signal *01.01 Motor speed rpm* is scaled and written to the fieldbus actual value. The scaling is as follows:

| 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 *Raw data*, the drive parameters *50.06 FBA act1 tr src* and *50.07 FBA act2 tr src* select the drive values for fieldbus actual value ACT1 and ACT2. The table below shows the scaling.

| Drive value | Corresponding fieldbus actual value ACT1 or ACT2 [integer] |
|---|---|
| k × 0.5<br>(k = value of parameter *58.11*) | 32 767 |
| 0 | 0 |
| k × -0.5<br>(k = value of parameter *58.11*) | -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 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 *339*. |
| 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 *341*. |
| 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 | Register address (16-bit drive parameter) = 400000 + 100 × group + index<br>Example: Modbus register address to drive parameter 03.18 is 400000 + 100 × 3 + 18 = 400318<br>Drive parameter access (32-bit drive parameter) = 420000 + 200 × group + 2 × index<br>Example: Modbus register address to drive parameter 01.27<br>420000 + 200 × 1 + 2 × 27 = 420254 |

## 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 *339*. |
| 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 *341*. |
| 400052 | Fieldbus actual value 1 (ACT1) |
| 400053 | Fieldbus actual value 2 (ACT2) |
| 40054 | Fieldbus data in/out 13 (Drive parameter *58.47 Data I/O 12*) |
| … | … |
| 40065 | Fieldbus data in/out 24 (Drive parameter *58.58 Data I/O 24* |
| 400101…409999 | Register address (16-bit drive parameter) = 400000 + 100 × group + index<br>Example: Modbus register address to drive parameter 03.18 is 400000 + 100 × 3 + 18 = 400318<br>Drive parameter access  (32-bit drive parameter) = 420000 + 200 × group + 2 × index<br>Example: Modbus register address to drive parameter 01.27<br>420000 + 200 × 1 + 2 × 27 = 420254 |

# 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 to15 (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 (SW) 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 *365* in chapter *Control through a fieldbus adapter*.

## References for the DCU 16-bit profile

See section *References for the ABB Drives profiles* on page *344*.

## Actual signals for the DCU 16-bit profile

See section *Actual values for the ABB Drives profiles* on page *345*.

## 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 *50.04 FBA ref1 modesel*) |
| 400053 | Actual value 2 (selected by parameter *50.05 FBA 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 | Register address (16-bit drive parameter) = 400000 + 100 × group + index<br>Example: Modbus register address to drive parameter 03.18 is 400000 + 100 × 3 + 18 = 400318<br>Drive parameter access (32-bit drive parameter) = 420000 + 200 × group + 2 × index<br>Example: Modbus register address to drive parameter 01.27<br>420000 + 200 × 1 + 2 × 27 = 420254 |

# 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 (SW).

## State transition diagram for the DCU 32-bit profile

See section *State diagram* on page *365* 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 FBA ref1 modesel* and *50.05 FBA 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. 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.

- If you select value *Speed*, the fieldbus reference can be used as a speed 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 speed reference in the drive [rpm] |
|---|---|
| 32767.65535 | 32767.65535 |
| 0 | 0 |
| -32768.65535 | -32768.65535 |

### 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 fieldbus 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 FBA ref1 modesel* and *50.05 FBA ref2 modesel* select the drive actual signals for the fieldbus actual values ACT1 and ACT2 respectively as follows:

- If you select value *Raw data*, the drive parameters *50.06 FBA act1 tr src* and *50.07 FBA act2 tr src* select the drive parameters for the fieldbus actual value

ACT1 and ACT2 respectively. 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 |

- If you select value *Speed*, the drive parameter *01.01 Motor speed rpm* will be written to fieldbus actual value. The table below clarifies the relation between the value of drive parameter value and the 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 *50.04 FBA ref1 modesel*) – Least significant 16-bits |
| 400054 | Actual value 1 (selected by parameter *50.04 FBA ref1 modesel*) – Most significant 16-bits |
| 400055 | Actual value 2 (selected by parameter *50.05 FBA ref2 modesel*) – Least significant 16-bits |
| 400056 | Actual value 2 (selected by parameter *50.05 FBA 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 | Register address (16-bit drive parameter) = 400000 + 100 × group + index<br>Example: Modbus register address to drive parameter 03.18 is 400000 + 100 × 3 + 18 = 400318<br>Drive parameter access (32-bit drive parameter) = 420000 + 200 × group + 2 × index<br>Example: Modbus register address to drive parameter 01.27<br>420000 + 200 × 1 + 2 × 27 = 420254 |

# 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 | 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:<br>00 Return Query Data:<br>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.<br>01 Restart Communications Option:<br>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.<br>04 Force Listen Only Mode:<br>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<br>Transport / Read Device Identification | Allows reading of identification and other information of the server.<br>Parameter "Read Device ID code" supports one access type:<br>01: Request to get the basic device identification. Returns ABB,ACQ810. |

## 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 in 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. |

10

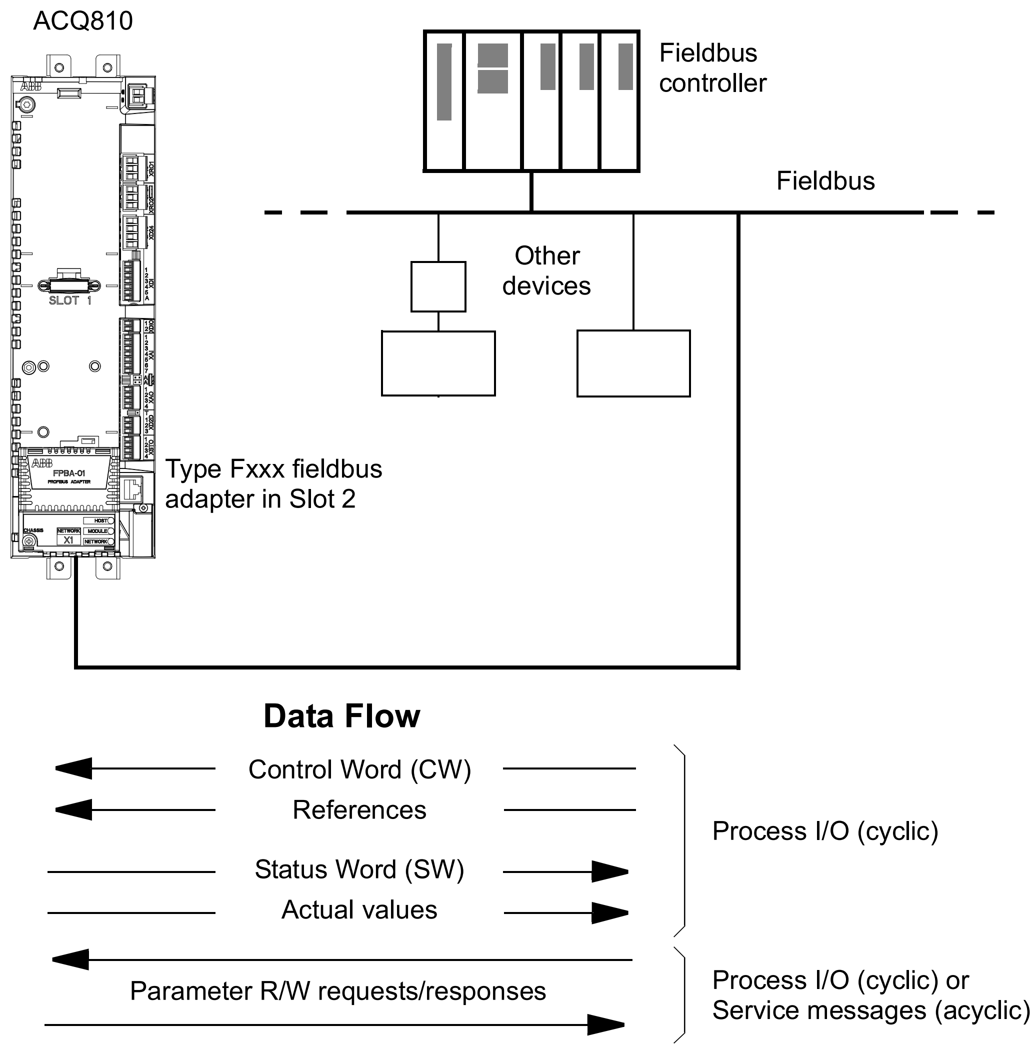
# Control through a fieldbus adapter

## What this chapter contains

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

## System overview

The drive can be connected to a fieldbus controller via a fieldbus adapter module. The adapter module is installed into drive Slot 2.




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 analog inputs.

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

- DeviceNet (FDNA-xx adapter)
- EtherNet/IP (FENA-xx adapter)
- LONWORKS® (FLON-xx adapter)
- Modbus (FSCA-xx adapter)
- PROFIBUS DP (FPBA-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 *229*) | | |
| | | |
| *50.01 FBA enable* | (1) *Enable* | Initialises communication between drive and fieldbus adapter module. |
| *50.02 Comm loss func* | (0) *No*<br>(1) *Fault*<br>(2) *Spd ref Safe*<br>(3) *Last speed* | 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 FBA ref1 modesel* and *50.05 FBA ref2 modesel* | (0) *Raw data*<br>(2) *Speed* | Defines the fieldbus reference scaling.<br>When *Raw data* is selected, see also parameters *50.06…50.11*. |
| | | |
| ADAPTER MODULE CONFIGURATION (see also page *231*) | | |
| | | |
| *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 *User's Manual* 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) *Done*<br>(1) *Refresh* | 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. |
| *51.29 Drive type code* | – | Displays the drive type code of the fieldbus adapter module mapping file stored in the memory of the drive. |

| Parameter | Setting for fieldbus control | Function/Information |
|---|---|---|
| *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 *User's Manual* 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 *232*) | | |
| | | |
| *52.01 FBA data in1* … *52.12 FBA data in12* | 4…6<br>14…16<br>101…9999 | Defines the data transmitted from drive to fieldbus controller.<br>**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<br>11…13<br>1001…9999 | Defines the data transmitted from fieldbus controller to drive.<br>**Note:** If the selected data is 32 bits long, two parameters are reserved for the transmission. |
| **Note:** In the *User's Manual* 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 *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 (before powering off the drive, wait at least 1 minute), or when parameter *51.27 FBA par refresh* is activated.

## Drive control parameters

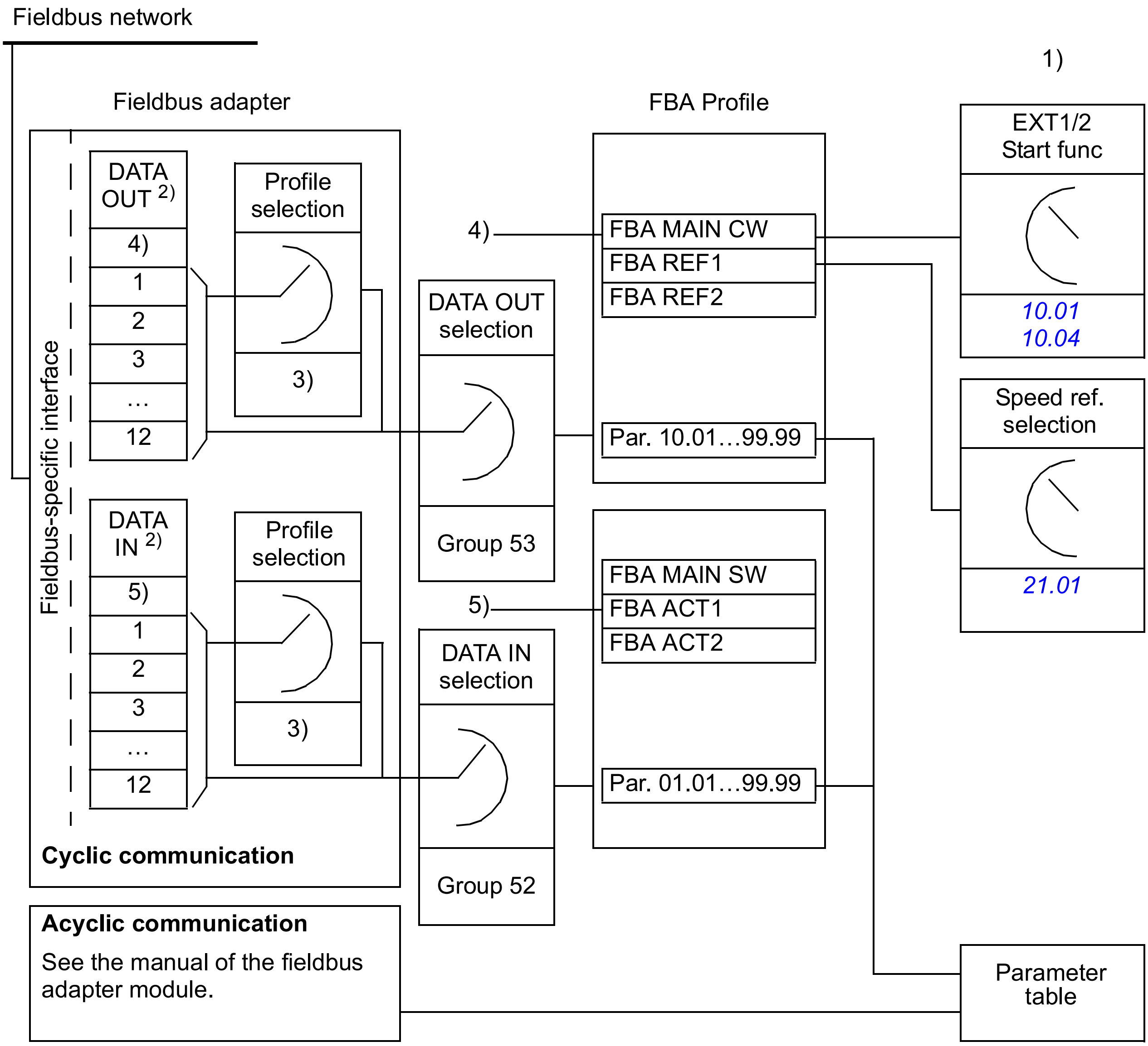
The Setting for fieldbus control column gives the value to use when the fieldbus interface is the desired source or destination for that particular signal. The Function/ Information column gives a description of the parameter.

| Parameter | Setting for fieldbus control | Function/Information |
|---|---|---|
| | | |
| CONTROL COMMAND SOURCE SELECTION | | |
| | | |
| *10.01 Ext1 start func* | (3) *FBA* | 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* | (3) *FBA* | Selects fieldbus as the source for the start and stop commands when EXT2 is selected as the active control location. |
| *21.01 Speed ref1 sel* | (3) *FBA ref1*<br>(4) *FBA ref2* | Fieldbus reference REF1 or REF2 is used as speed reference. |
| | | |
| SYSTEM CONTROL INPUTS | | |
| | | |
| *16.07 Param save* | (0) *Done*<br>(1) *Save* | Saves parameter value changes (including those made through fieldbus control) to permanent memory. |

# The fieldbus control 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.

### The Control Word and the 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 *365* presents the most important states (including the FBA profile state names). The FBA Control Word (parameter *02.24* – see page *117*) commands the transitions between these states and the FBA Status Word (parameter *02.26* – see page *118*) 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, and ABB Drives profile for all fieldbus adapter modules) 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

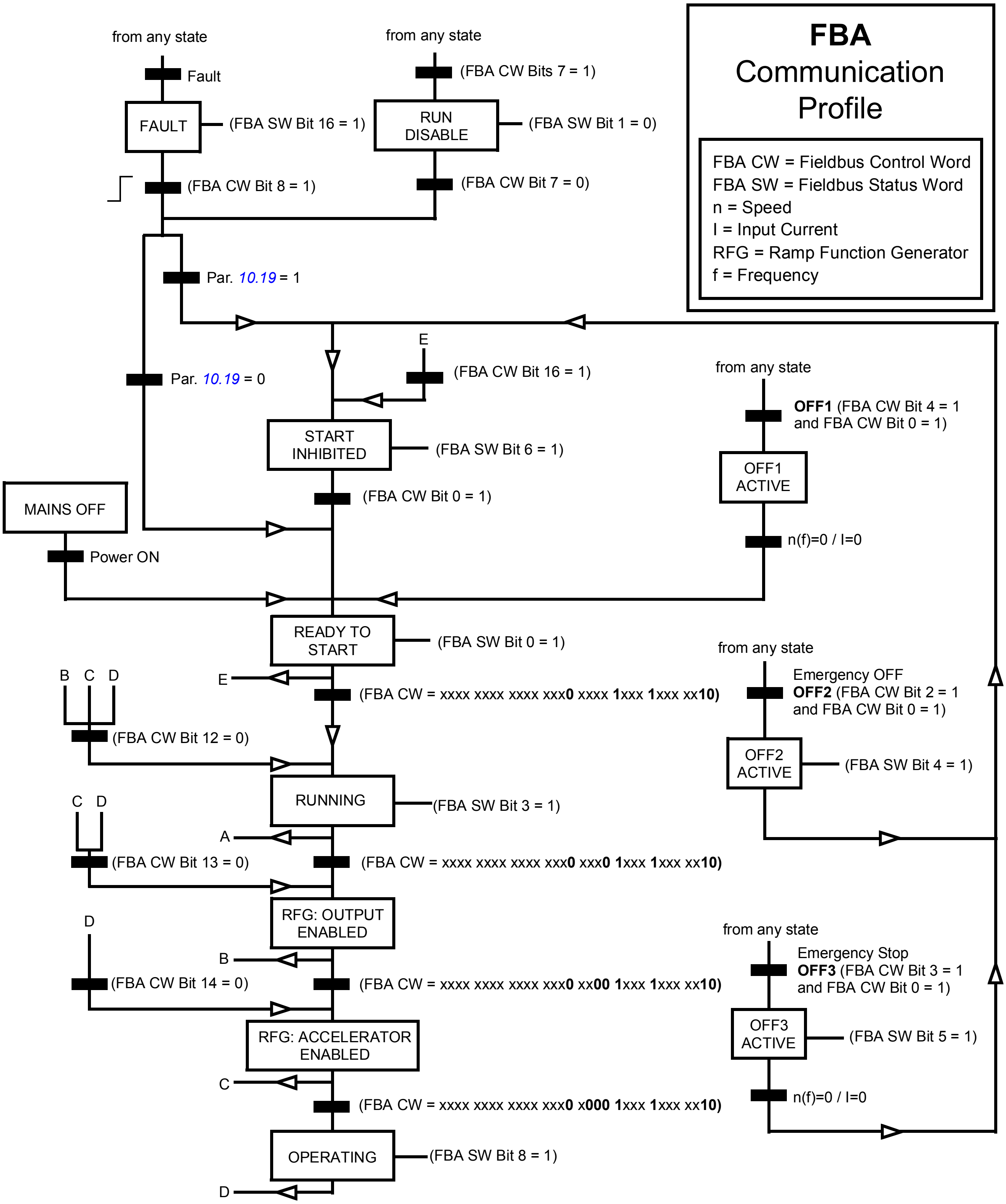
References (FBA REF) are 16/32-bit signed integers. A negative reference (indicating reversed direction of rotation) is formed by calculating the two’s complement from the corresponding positive reference value. The contents of each reference word can be used as torque or speed reference.

When torque or speed reference scaling is selected (by parameter *50.04 FBA ref1 modesel* / *50.05 FBA 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.



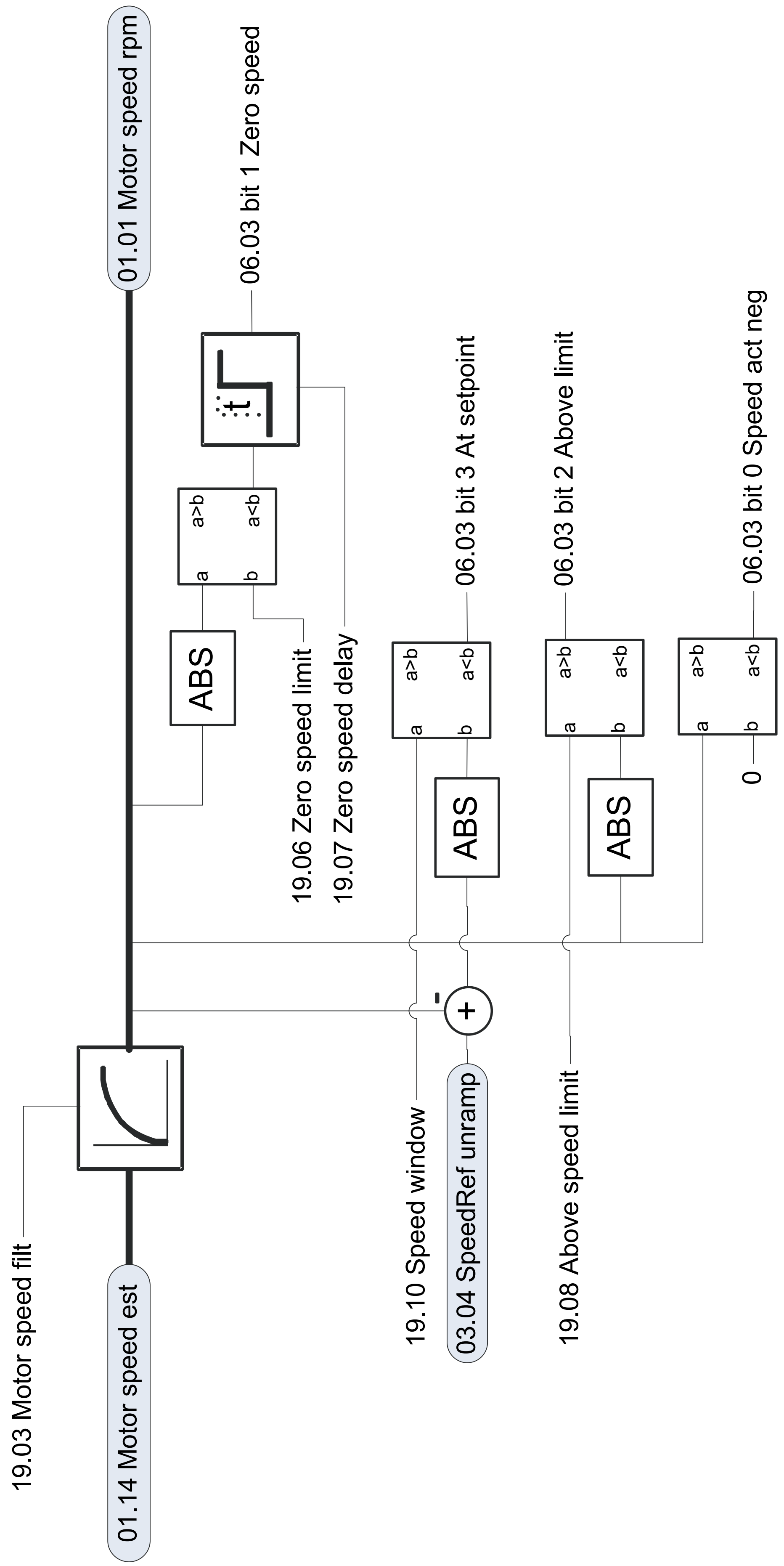
11

# Control block diagrams

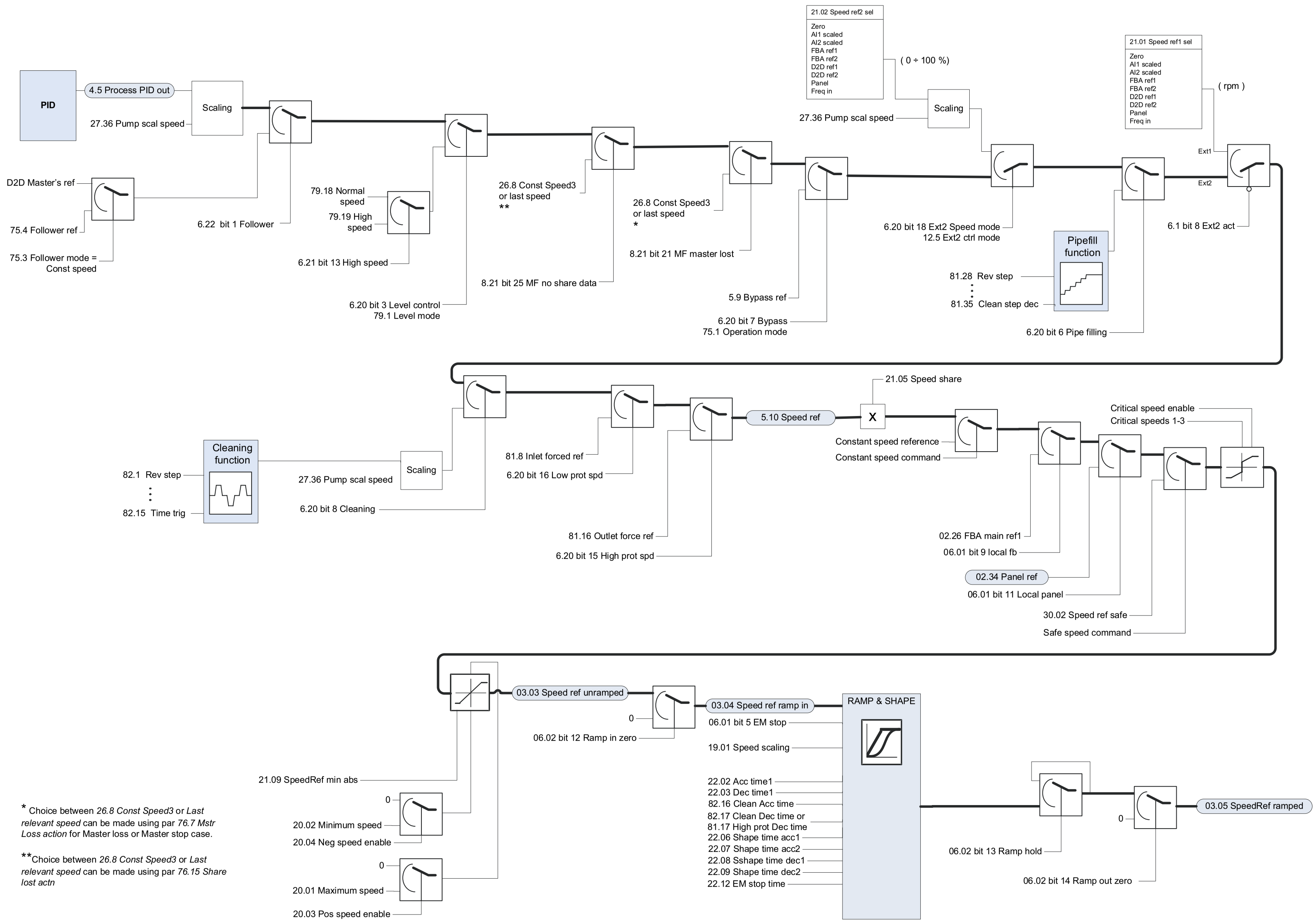
## What this chapter contains

The chapter contains a graphical representation of the control program.

## Speed feedback



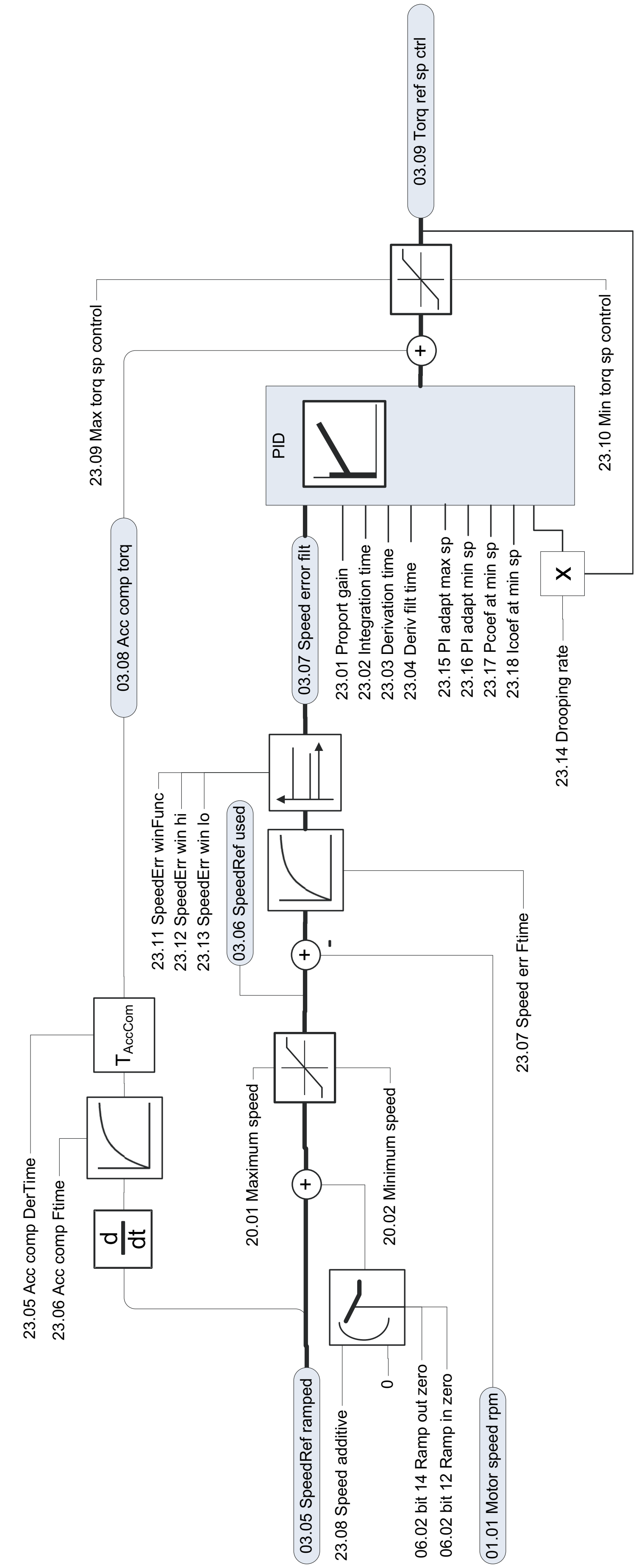
# Speed reference chain



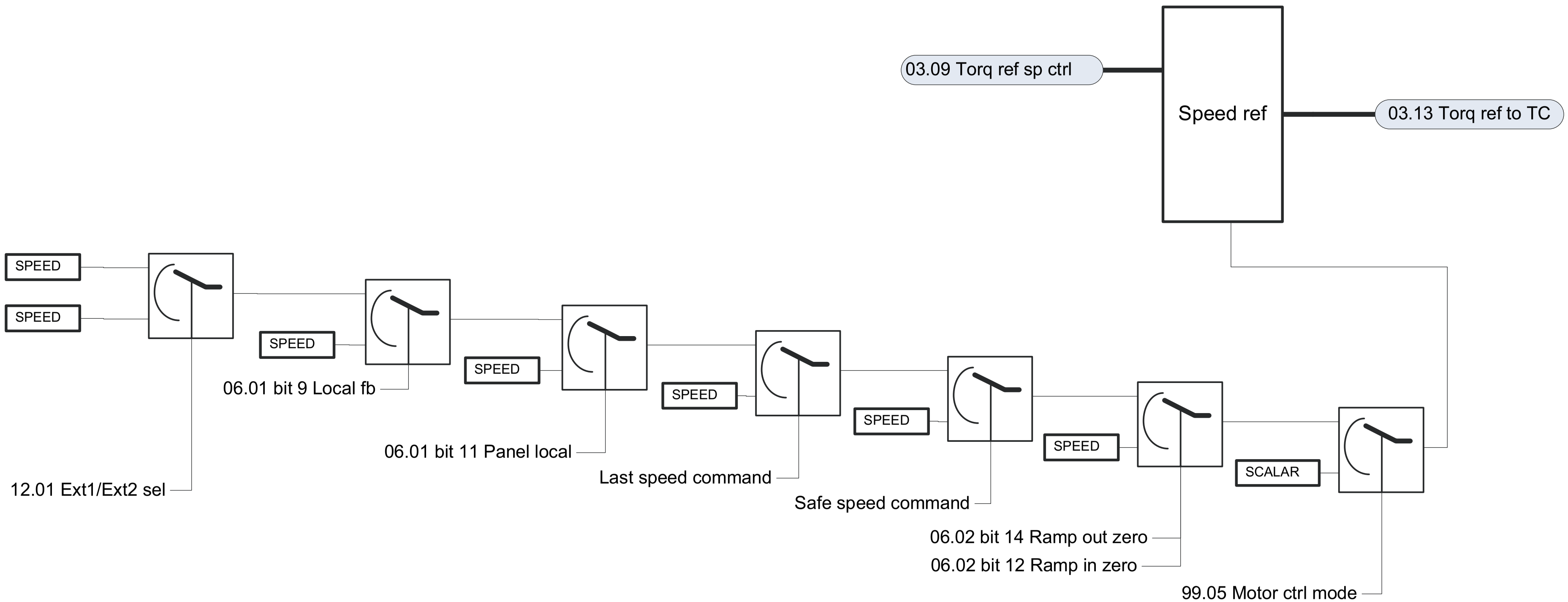

* Choice between *26.8 Const Speed3* or *Last relevant speed* can be made using par *76.7 Mstr Loss action* for Master loss or Master stop case.

**Choice between *26.8 Const Speed3* or *Last relevant speed* can be made using par *76.15 Share lost actn*

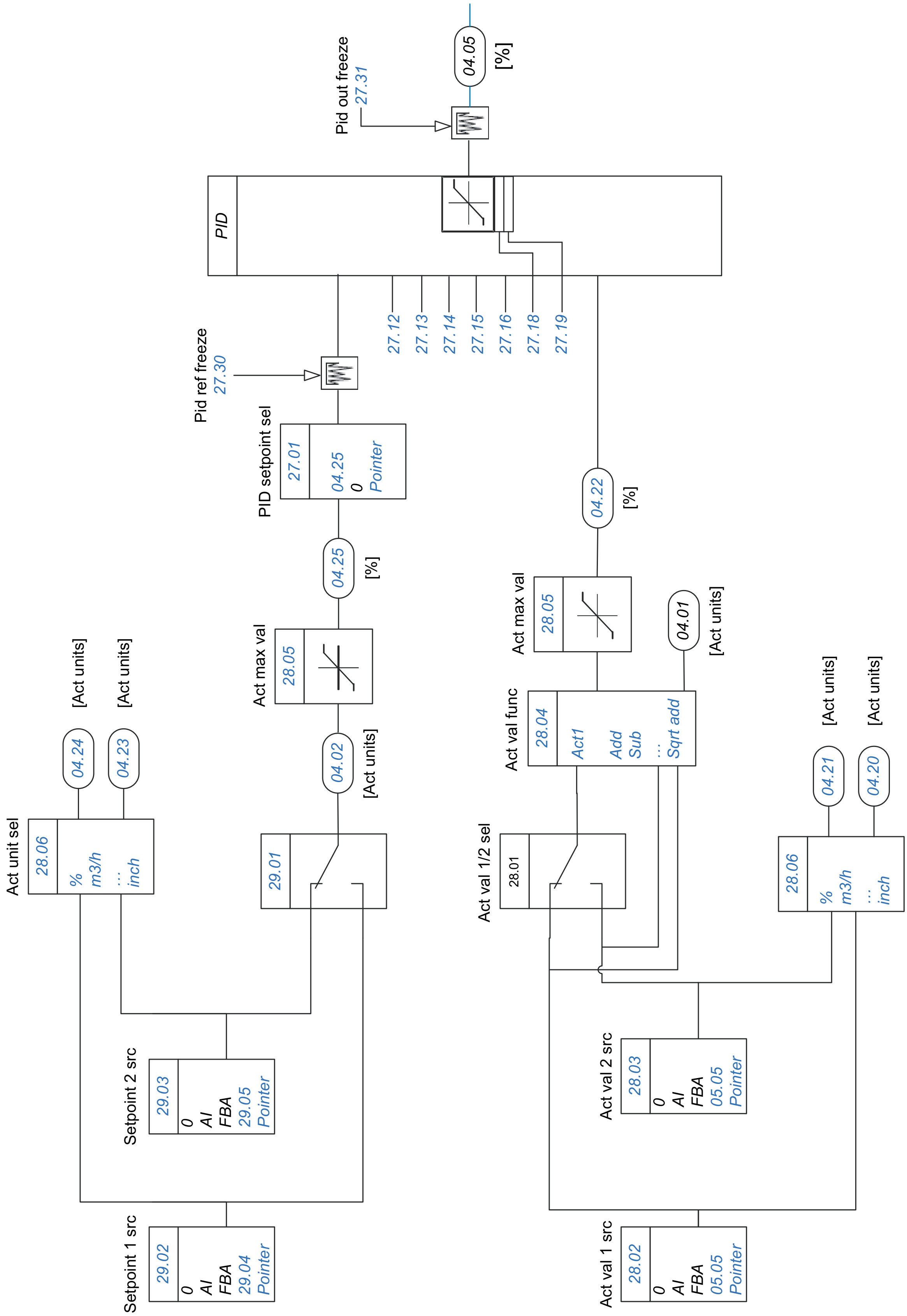
# Speed error handling



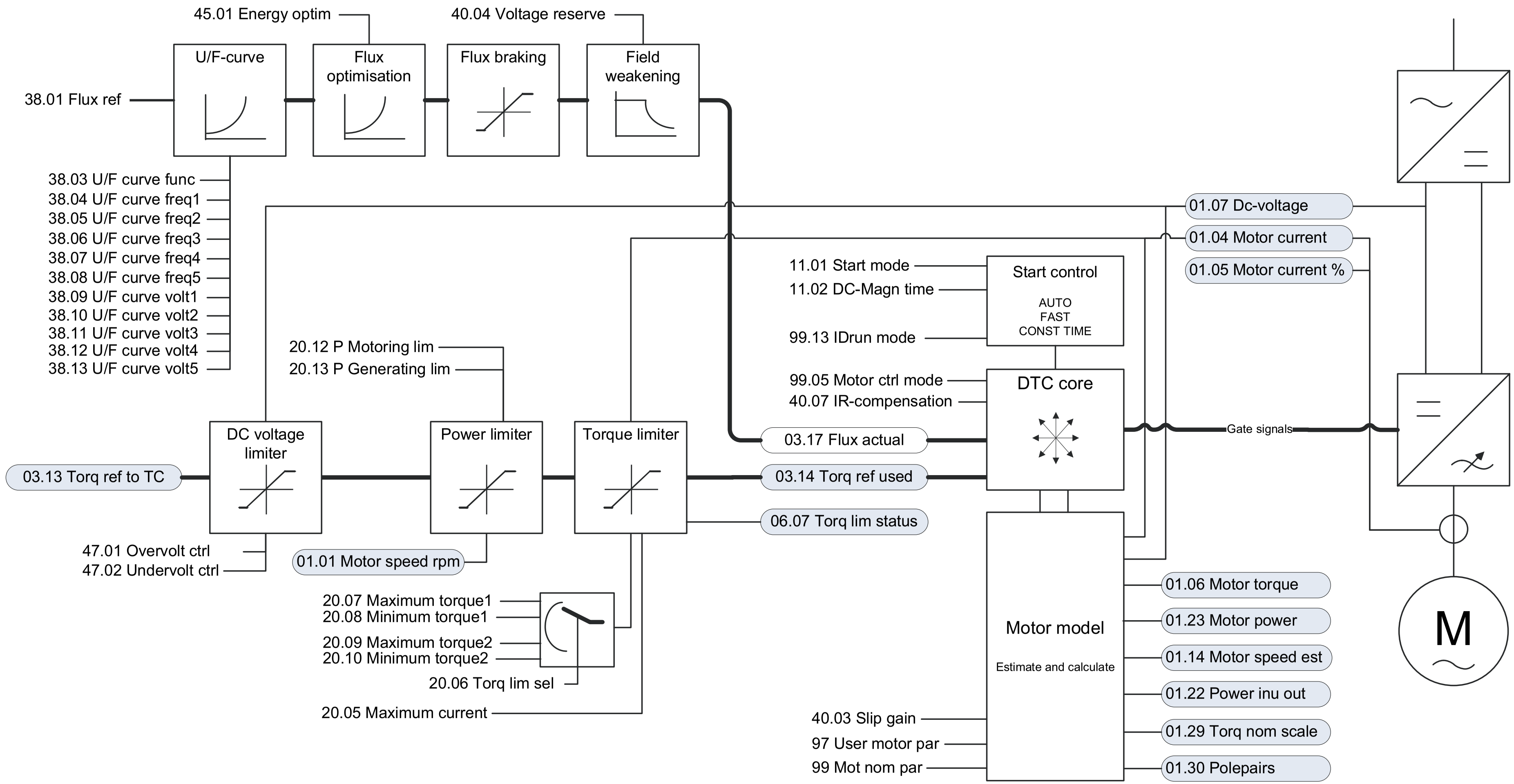
# Torque reference modification, operating mode selection



# Process PID control setpoint and actual value selection



# Direct torque control



# 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.

3AUA0000055144 Rev B / EN
EFFECTIVE: 2011-02-14

**ABB Oy**
AC Drives
P.O. Box 184
FI-00381 HELSINKI
FINLAND
Telephone +358 10 22 11
Fax +358 10 22 22681
Internet http://www.abb.com

**ABB Inc.**
Automation Technologies
Drives & Motors
16250 West Glendale Drive
New Berlin, WI 53151 USA
Telephone 262 785-3200
800-HELP-365
Fax 262 780-5135

**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
Internet http://www.abb.com

ABB low voltage AC drives

# Application guide
# Application programming for ACS850 and ACQ810 drives

Power and productivity
for a better world™

# List of related manuals

| Drive hardware manuals and guides | Code (English) | |
|---|---|---|
| *ACS850-04 (0.37…45 kW) hardware manual* | 3AUA0000045496 | 1) |
| *ACS850-04 (0.37…45 kW) quick installation guide* | 3AUA0000045495 | 1) |
| *ACS850-04 (55…160 kW, 75…200 hp) hardware manual* | 3AUA0000045487 | 1) |
| *ACS850-04 (55…160 kW, 75…200 hp) quick installation guide* | 3AUA0000045488 | 1) |
| *ACS850-04 (200…500 kW, 250…600 hp) hardware manual* | 3AUA0000026234 | 1) |
| *ACS850-04 (400…560 kW, 450…700 hp) hardware manual* | 3AUA0000081249 | 1) |
| *ACQ810-04 (0.37…45 kW, 0.5…60 hp) hardware manual* | 3AUA0000055160 | 1) |
| *ACQ810-04 (55…160 kW, 75…200 hp) hardware manual* | 3AUA0000055161 | 1) |
| *ACQ810-04 (200…400 kW, 250…600 hp) hardware manual* | 3AUA0000055155 | 1) |

| Drive firmware manuals and guides | | |
|---|---|---|
| *ACS850 standard control program quick start-up guide* | 3AUA0000045498 | 2) |
| *ACS850 standard control program firmware manual* | 3AUA0000045497 | 3) |
| *ACQ810-04 drive modules start-up guide* | 3AUA0000055159 | 2) |
| *ACQ810 standard pump control program firmware manual* | 3AUA0000055144 | 3) |

| Option manuals and guides | | |
|---|---|---|
| *ACS-CP-U control panel IP54 mounting platform kit (+J410) installation guide* | 3AUA0000049072 | 1) |
| *Manuals and quick guides for I/O extension modules, fieldbus adapters, etc.* | | 1) |

| Drive PC tools manuals | |
|---|---|
| *DriveSPC user manual* | 3AFE68836590 |
| *DriveStudio user manual* | 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 and ACQ810 drives

Table of contents



3AUA0000078664 Rev B
EN
EFFECTIVE: 2011-11-28

# Table of contents

*5. Examples of using standard function blocks*



*Further information*

1

# 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 and ACQ810 drives with the Standard pump 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 and ACQ810 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

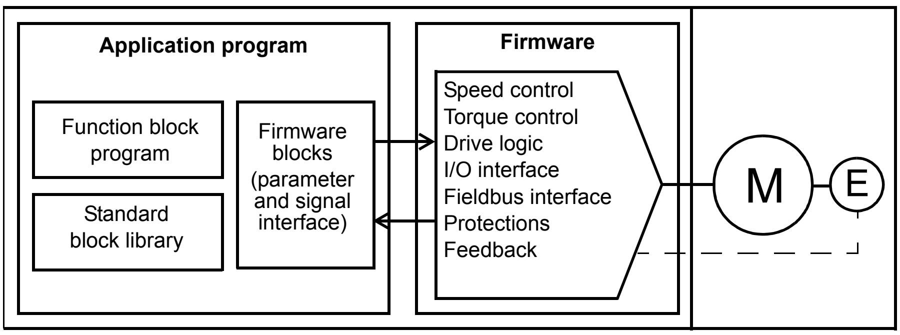
This chapter introduces drive programming and describes application programming using the DriveSPC PC tool.

## General about drive programming

The drive control program is divided into two parts:

- firmware program
- application program.

**Drive control program**

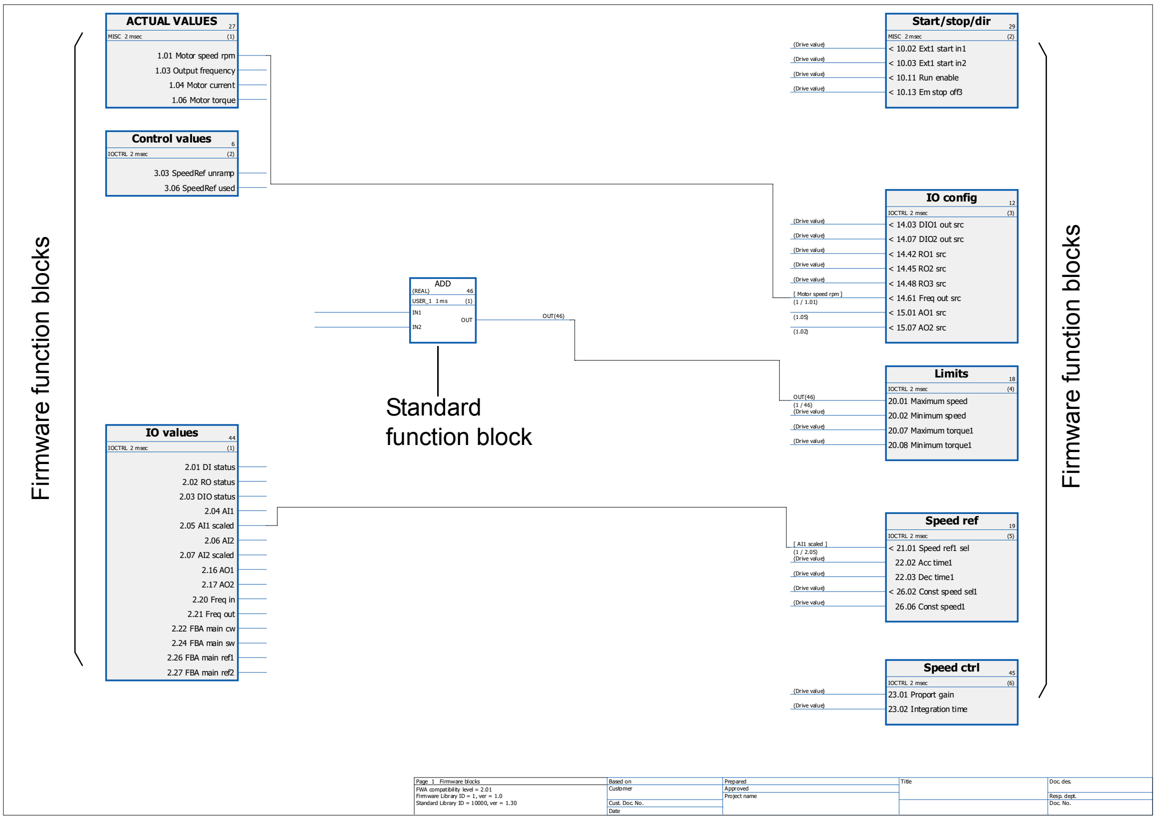



The firmware program performs the main control functions, including speed and torque control, drive logic (start/stop), I/O, feedback, communication and protection functions. Firmware functions are configured and programmed with parameters. Parameters can be set via the drive control panel, the DriveStudio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

3

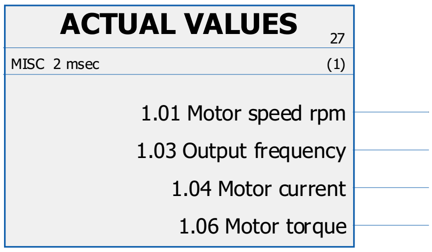
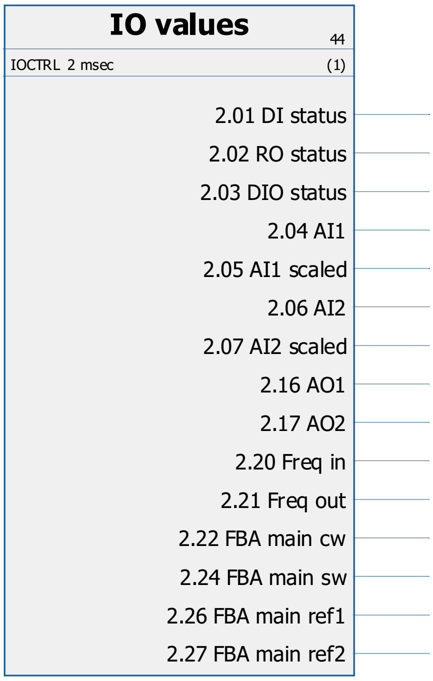
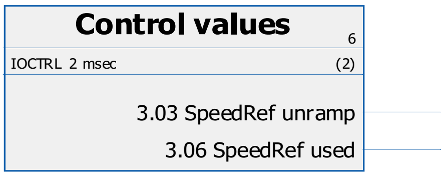
# Firmware function blocks

## What this chapter contains

This chapter presents the firmware function blocks. The blocks are grouped according to parameter numbering in the drive firmware.

**Note:** The Speed ctrl block is not available with ACQ810 drives.

**Note:** Parameter 14.48 RO3 src does not exist with ACQ810 drives.

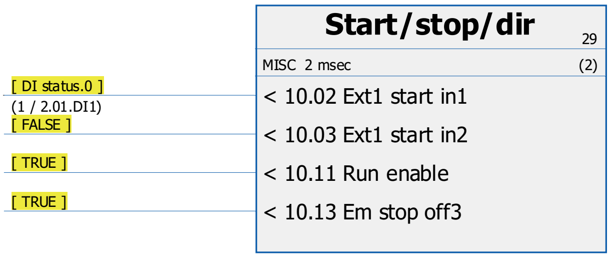
| Description | Illustration |
|---|---|
| **ACTUAL VALUES**<br>(1) | |
| Basic signals for monitoring the drive. |  |
| **IO values**<br>(2) | |
| Input and output signals. |  |
| **Control values**<br>(3) | |
| Speed reference values. |  |

| Description | Illustration |
|---|---|

## Start/stop/dir

(10)

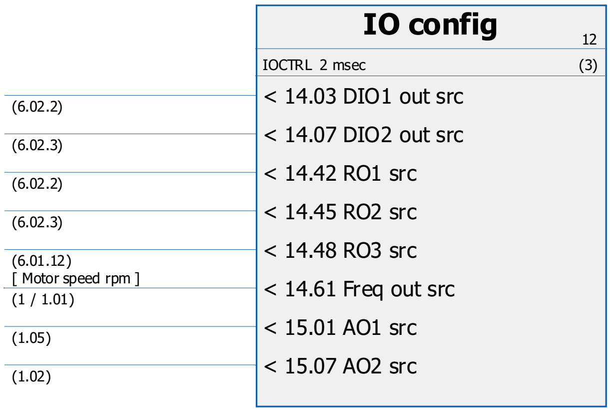
Source selections for start/stop, run enable and emergency stop signals.




## IO config

(13)

Configuration of digital input/outputs, relay outputs and analog outputs.

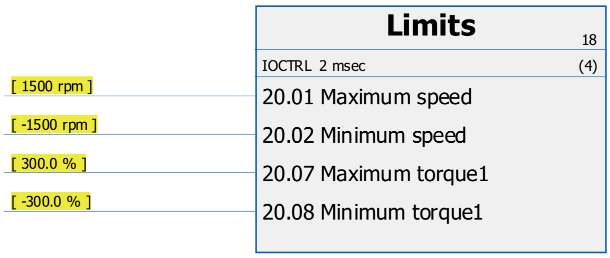


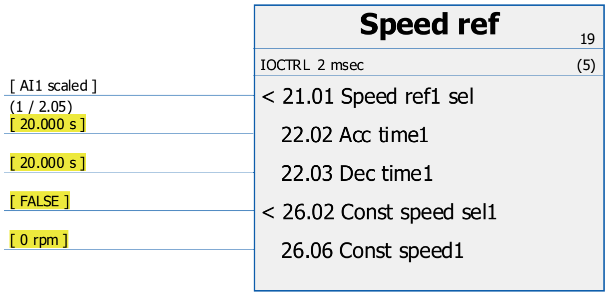
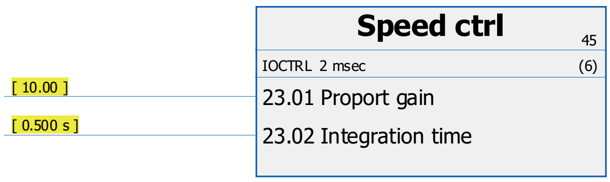

**Note:** Parameter 14.48 RO3 src does not exist with ACQ810 drives.

## Limits

(20)

Drive operation limits.



| Description | Illustration |
|---|---|
| **Speed ref**<br>(21) | |
| Speed reference source selection; acceleration/ deceleration and constant speed settings. |  |
| **Speed ctrl**<br>(23) | |
| Speed controller settings. | <br><br>**Note:** This block is not available with ACQ810 drives. |

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.

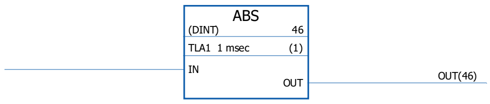
**Note:** The given execution times can vary depending on the drive application used.

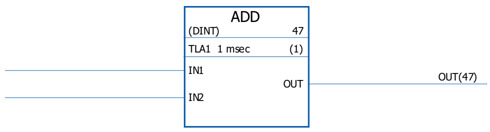
## Terms

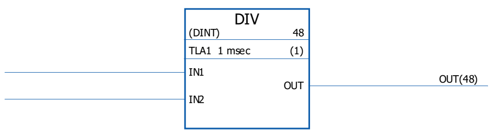
| Data type | Description | Range |
|---|---|---|
| Boolean | Boolean | 0 or 1 |
| DINT | 32-bit integer value (31 bits + sign) | -2147483648…2147483647 |
| INT | 16-bit integer value (15 bits + sign) | -32768…32767 |
| PB | Packed Boolean | 0 or 1 for each individual bit |
| REAL | $\underbrace{\text{16-bit value}}_{=\text{ integer value}}$ $\underbrace{\text{16-bit value}}_{=\text{ fractional value}}$ (31 bits + sign) | -32768,99998…32767,9998 |
| REAL24 | $\underbrace{\text{8-bit value}}_{=\text{ integer value}}$ $\underbrace{\text{24-bit value}}_{=\text{ fractional value}}$ (31 bits + sign) | -128,0…127,999 |

# Alphabetical index

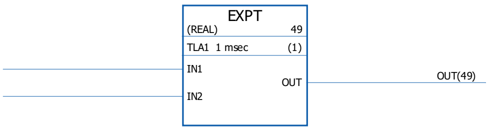
# Arithmetic

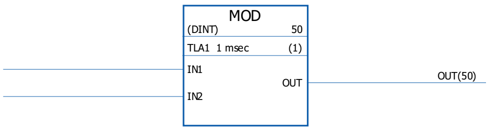
| **ABS**<br>(10001) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.53 µs |
| **Operation** | The output (OUT) is the absolute value of the input (IN).<br>OUT = \| IN \| |
| **Inputs** | The input data type is selected by the user.<br>Input (IN): DINT, INT, REAL or REAL24 |
| **Outputs** | Output (OUT): DINT, INT, REAL or REAL24 |

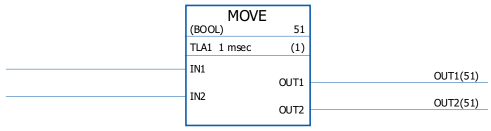
| **ADD**<br>(10000) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 3.36 µs (when two inputs are used) + 0.52 µs (for every additional input). When all inputs are used, the execution time is 18.87 µs. |
| **Operation** | The output (OUT) is the sum of the inputs (IN1…IN32).<br>OUT = IN1 + IN2 + … + IN32<br>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.<br>Input (IN1…IN32): DINT, INT, REAL or REAL24 |
| **Outputs** | Output (OUT): DINT, INT, REAL or REAL24 |

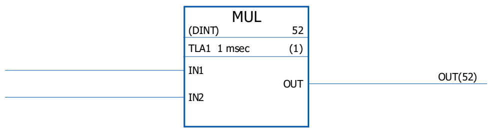
| **DIV**<br>(10002) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 2.55 µs |
| **Operation** | The output (OUT) is input IN1 divided by input IN2.<br>OUT = IN1/IN2<br>The output value is limited to the maximum and minimum values defined by the selected data type range.<br>If the divider (IN2) is 0, the output is 0. |

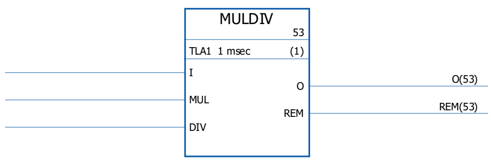
| | |
|---|---|
| **Inputs** | The input data type is selected by the user.<br>Input (IN1, IN2): INT, DINT, REAL, REAL24 |
| **Outputs** | Output (OUT): INT, DINT, REAL, REAL24 |

| **EXPT**<br>(10003) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 81.90 ìs |
| **Operation** | The output (OUT) is input IN1 raised to the power of the input IN2:<br>OUT = IN1IN2<br>If input IN1 is 0, the output is 0.<br>The output value is limited to the maximum value defined by the selected data type range.<br>**Note:** The execution of the EXPT function is slow. |
| **Inputs** | The input data type is selected by the user.<br>Input (IN1): REAL, REAL24<br>Input (IN2): REAL |
| **Outputs** | Output (OUT): REAL, REAL24 |

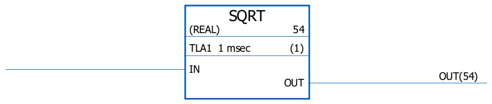
| **MOD**<br>(10004) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 1.67 µs |
| **Operation** | The output (OUT) is the remainder of the division of the inputs IN1 and IN2.<br>OUT = remainder of IN1/IN2<br>If input IN2 is zero, the output is zero. |
| **Inputs** | The input data type is selected by the user.<br>Input (IN1, IN2): INT, DINT |
| **Outputs** | Output (OUT): INT, DINT |

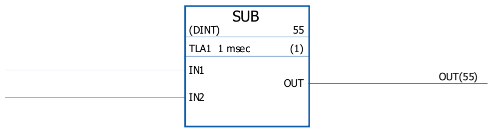
| **MOVE**<br>(10005) | |
|---|---|
| **Illustration** |   |
| **Execution time** | 2.10 µs (when two inputs are used) + 0.42 µs (for every additional input). When all inputs are used, the execution time is 14.55 µs. |
| **Operation** | Copies the input values (IN1…32) to the corresponding outputs (OUT1…32). |
| **Inputs** | The input data type and number of inputs (2…32) are selected by the user.<br>Input (IN1…IN32): INT, DINT, REAL, REAL24, Boolean |
| **Outputs** | Output (OUT1…OUT32): INT, DINT, REAL, REAL24, Boolean |

| **MUL**<br>(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).<br>O = IN1 × IN2 × … × IN32<br>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.<br>Input (IN1…IN32): INT, DINT, REAL, REAL24 |
| **Outputs** | Output (OUT): INT, DINT, REAL, REAL24 |

| **MULDIV**<br>(10007) | |
|---|---|
| **Illustration** |   |
| **Execution time** | 7.10 µs |

| | |
|---|---|
| **Operation** | The output (O) is the product of input IN and input MUL divided by input DIV.<br>Output = (I × MUL) / DIV<br>O = whole value. REM = remainder value.<br>Example: I = 2, MUL = 16 and DIV = 10:<br>(2 × 16) / 10 = 3.2, i.e. O = 3 and REM = 2<br>The output value is limited to the maximum and minimum values defined by the data type range. |
| **Inputs** | Input (I): DINT<br>Multiplier input (MUL): DINT<br>Divider input (DIV): DINT |
| **Outputs** | Output (O): DINT<br>Remainder output (REM): DINT |

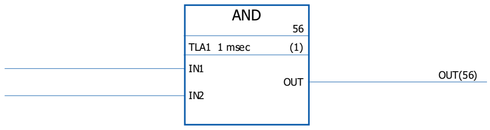
| **SQRT**<br>(10008) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 2.09 µs |
| **Operation** | Output (OUT) is the square root of the input (IN).<br>OUT = sqrt(IN)<br>Output is 0 if the input value is negative |
| **Inputs** | The input data type is selected by the user.<br>Input (IN): REAL, REAL24 |
| **Outputs** | Output (OUT): REAL, REAL24 |

| **SUB**<br>(10009) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 2.33 µs |
| **Operation** | Output (OUT) is the difference between the input signals (IN):<br>OUT = IN1 - IN2<br>The output value is limited to the maximum and minimum values defined by the selected data type range. |
| **Inputs** | The input data type is selected by the user.<br>Input (IN1, IN2): INT, DINT, REAL, REAL24 |
| **Outputs** | Output (OUT): INT, DINT, REAL, REAL24 |

# Bitstring

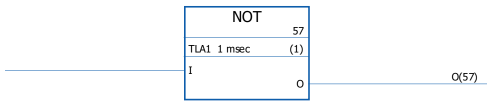
## AND

(10010)

<table>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>1.55 µs (when two inputs are used) + 0.60 µs (for every additional input). When all inputs are used, the execution time is 19.55 µs.</td></tr>
<tr><td>Operation</td><td>The output (OUT) is 1 if all the connected inputs (IN1…IN32) are 1. Otherwise the output is 0.<br>Truth table:
<table>
<tr><th>IN1</th><th>IN2</th><th>OUT</th></tr>
<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>
</table>
The inputs can be inverted.</td></tr>
<tr><td>Inputs</td><td>The number of inputs is selected by the user.<br>Input (IN1…IN32): Boolean</td></tr>
<tr><td>Outputs</td><td>Output (OUT): Boolean</td></tr>
</table>

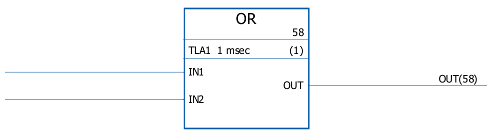
## NOT

(10011)

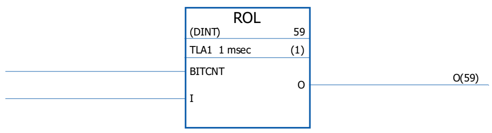
<table>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>0.32 µs</td></tr>
<tr><td>Operation</td><td>The output (O) is 1 if the input (I) is 0. The output is 0 if the input is 1.</td></tr>
<tr><td>Inputs</td><td>Input (I): Boolean</td></tr>
<tr><td>Outputs</td><td>Output (O): Boolean</td></tr>
</table>

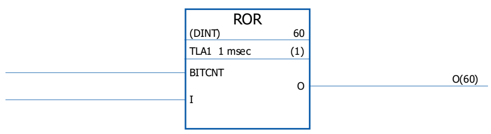
## OR

(10012)

<table>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>1.55 µs (when two inputs are used) + 0.60 µs (for every additional input). When all inputs are used, the execution time is 19.55 µs.</td></tr>
</table>

<table>
<tr><td>Operation</td><td>The output (OUT) is 0, if all connected inputs (IN) are 0. Otherwise the output is 1.<br>Truth table:<br>
<table>
<tr><th>IN1</th><th>IN2</th><th>OUT</th></tr>
<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>
</table>
The inputs can be inverted.</td></tr>
<tr><td>Inputs</td><td>The number of inputs (2…32) is selected by the user.<br>Input (IN1…IN32): Boolean</td></tr>
<tr><td>Outputs</td><td>Output (OUT): Boolean</td></tr>
</table>

<table>
<tr><th colspan="2">ROL<br>(10013)</th></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>1.28 µs</td></tr>
<tr><td>Operation</td><td>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.<br>Example: If BITCNT = 3<br>3 MSB
<table>
<tr><td>I</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>O</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>
</table>
3 LSB</td></tr>
<tr><td>Inputs</td><td>The input data type is selected by the user.<br>Number of bits input (BITCNT): INT, DINT<br>Input (I): INT, DINT</td></tr>
<tr><td>Outputs</td><td>Output (O): INT, DINT</td></tr>
</table>

<table>
<tr><th colspan="2">ROR<br>(10014)</th></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>1.28 µs</td></tr>
</table>

**Operation**

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.

Example: If BITCNT = 3

3 LSB

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **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** | 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 |

3 MSB

**Inputs**

The input data type is selected by the user.

Number of bits input (BITCNT): INT, DINT

Input (I): INT, DINT

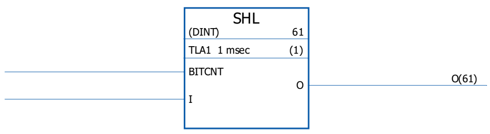
**Outputs**

Output (O): INT, DINT

## SHL

(10015)

**Illustration**




**Execution time**

0.80 µs

**Operation**

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.

Example: If BITCNT = 3

3 MSB

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **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 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |

3 LSB

**Inputs**

The input data type is selected by the user.

Number of bits (BITCNT): INT; DINT

Input (I): INT, DINT

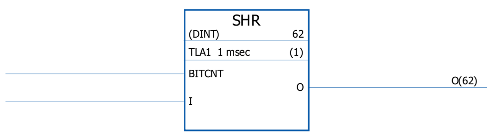
**Outputs**

Output (O): INT; DINT

## SHR

(10016)

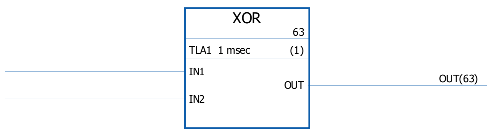
**Illustration**



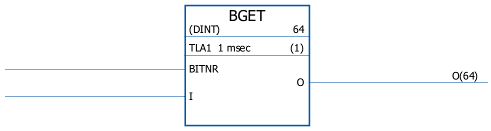

**Execution time**

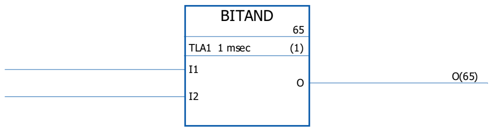
0.80 µs

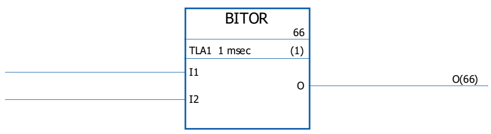
<table>
<tr><td>Operation</td><td>Input bits (I) are rotated to the right by the number (N) of bits defined by BITCNT. The N least significant bits (LSB) of the input are lost and the N most significant bits (MSB) of the output are set to 0.<br>Example: If BITCNT = 3<br>3 LSB<table><tr><td>I</td><td>1 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>O</td><td>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</td></tr></table>3 MSB</td></tr>
<tr><td>Inputs</td><td>The input data type is selected by the user.<br>Number of bits (BITCNT): INT; DINT<br>Input (I): INT, DINT</td></tr>
<tr><td>Outputs</td><td>Output (O): INT; DINT</td></tr>
</table>

<table>
<tr><td colspan="2"><h2>XOR</h2>(10017)</td></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>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.</td></tr>
<tr><td>Operation</td><td>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.<br>Example:<table><tr><th>IN1</th><th>IN2</th><th>OUT</th></tr><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></table>The inputs can be inverted.</td></tr>
<tr><td>Inputs</td><td>The number of inputs (2…32) is selected by the user.<br>Input (IN1…IN32): Boolean</td></tr>
<tr><td>Outputs</td><td>Output (OUT): Boolean</td></tr>
</table>

# Bitwise

| **BGET**<br>(10034) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.88 µs |
| **Operation** | The output (O) is the value of the selected bit (BITNR) of the input (I).<br>BITNR: Bit number (0 = bit number 0, 31 = bit number 31)<br>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.<br>Number of the bit (BITNR): DINT<br>Input (I): DINT, INT |
| **Outputs** | Output (O): Boolean |

<table>
<tr><th colspan="2">BITAND<br>(10035)</th></tr>
<tr><td>Illustration</td><td></td></tr>
<tr><td>Execution time</td><td>0.32 µs</td></tr>
<tr><td>Operation</td><td>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.<br>Example:
<table>
<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>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>
</table></td></tr>
<tr><td>Inputs</td><td>Input (I1, I2): DINT</td></tr>
<tr><td>Outputs</td><td>Output (O): DINT</td></tr>
</table>

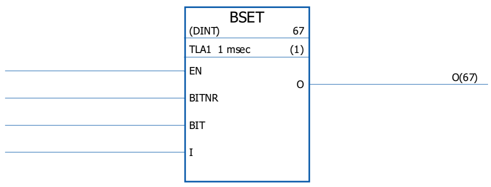
| **BITOR**<br>(10036) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.32 µs |

<table>
<tr><td>Operation</td><td>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.<br>Example:
<table>
<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>
</td></tr>
<tr><td>Inputs</td><td>Input (I1, I2): DINT</td></tr>
<tr><td>Outputs</td><td>Output (O): DINT</td></tr>
</table>

## BSET

(10037)

**Illustration**

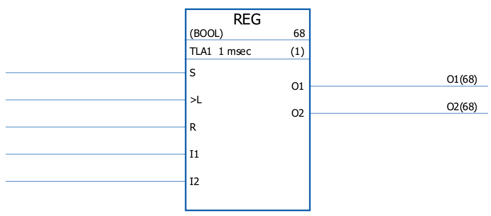



| | |
|---|---|
| **Execution time** | 1.36 µs |
| **Operation** | The value of a selected bit (BITNR) of the input (I) is set as defined by the bit value input (BIT). The function must be enabled by the enable input (EN).<br>BITNR: Bit number (0 = bit number 0, 31 = bit number 31)<br>If BITNR is not 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).<br>Example:<br>EN = 1, BITNR = 3, BIT = 0<br>IN = 0000 0000 1111 1111<br>O = 0000 0000 1111 0111 |
| **Inputs** | The input data type is selected by the user.<br>Enable input (EN): Boolean<br>Number of the bit (BITNR): DINT<br>Bit value input (BIT): Boolean<br>Input (I): INT, DINT |
| **Outputs** | Output (O): INT, DINT |

## REG

(10038)

**Illustration**



| | |
|---|---|
| **Execution time** | 2.27 µs (when two inputs are used) + 1.02 µs (for every additional input). When all inputs are used, the execution time is 32.87 µs. |

**Operation**

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.

If the reset input (R) is 1, all connected outputs are 0.

Example:

| 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 |

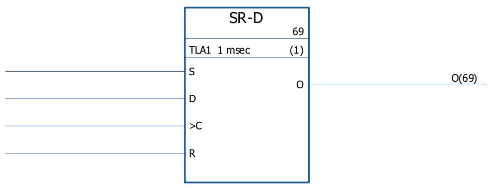
$O1_{previous}$ is the previous cycle output value.

| | |
|---|---|
| **Inputs** | The input data type and number of inputs (1…32) are selected by the user.<br>Set input (S): Boolean<br>Load input (L): Boolean<br>Reset input (R): Boolean<br>Input (I1…I32): Boolean, INT, DINT, REAL, REAL24 |
| **Outputs** | Output (O1…O32): Boolean, INT, DINT, REAL, REAL24 |

## SR-D

(10039)

**Illustration**




| | |
|---|---|
| **Execution time** | 1.04 µs |

<table>
<tr><td>Operation</td><td>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.<br>If only set (S) and reset (R) inputs are used, SR-D block acts as an SR block:<br>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.<br>Truth table:
<table>
<tr><th>S</th><th>R</th><th>D</th><th>C</th><th>$O_{previous}$</th><th>O</th></tr>
<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>
<tr><td colspan="6">$O_{previous}$ is the previous cycle output value.</td></tr>
</table>
</td></tr>
<tr><td>Inputs</td><td>Set input (S): Boolean<br>Data input (D): Boolean<br>Clock input (C): Boolean<br>Reset input (R): Boolean</td></tr>
<tr><td>Outputs</td><td>Output (O): Boolean</td></tr>
</table>

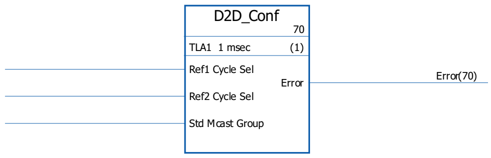
# Communication (ACS850 only)

See also section *Drive-to-drive communication (ACS850 only)* on page 95.

## D2D_Conf

(10092)

**Illustration**




**Execution time**: -

**Operation**

Defines handling interval for drive-to-drive references 1 and 2, and the address (group number) for standard (non-chained) multicast messages.

The values of the Ref1/2 Cycle Sel inputs correspond to the following intervals:

| Value | Handling interval |
|---|---|
| 0 | Default (500 µs for reference 1; 2 ms for reference 2) |
| 1 | 250 µs |
| 2 | 500 µs |
| 3 | 2 ms |

**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).

Allowable values for the Std Mcast Group input are 0 (= multicasting not used) and 1…62 (multicast group).

An unconnected input, or an input in an error state, is interpreted as having the value 0.

The error codes indicated by the Error output are as follows:

| 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**

Drive-to-drive reference 1 handling interval (Ref1 Cycle Sel): INT

Drive-to-drive reference 2 handling interval (Ref2 Cycle Sel): INT

Standard multicast address (Std Mcast Group): INT

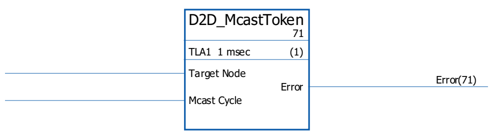
**Outputs**

Error output (Error): PB

## D2D_McastToken

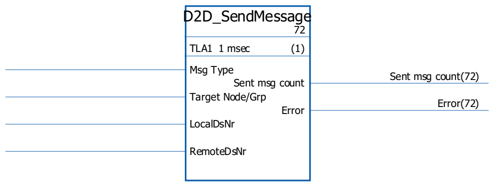
(10096)

**Illustration**




**Execution time**: -

<table>
<tr><td>Operation</td><td>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.<br>Note: This block is only supported in the master.<br>The Target Node input defines the node address the master sends the tokens to; the range is 1…62.<br>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.<br>The error codes indicated by the Error output are as follows:
<table>
<tr><th>Bit</th><th>Description</th></tr>
<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>
</table></td></tr>
<tr><td>Inputs</td><td>Token recipient (Target Node): INT<br>Token interval (Mcast Cycle): INT</td></tr>
<tr><td>Outputs</td><td>Error output (Error): DINT</td></tr>
</table>

<table>
<tr><td colspan="2">D2D_SendMessage<br>(10095)</td></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>-</td></tr>
</table>

**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 | Master P2P:<br>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).<br>The follower replies by sending the next dataset (RemoteDsNr + 1) to the master (LocalDsNr + 1).<br>The node number of a drive is defined by parameter *57.03*.<br>**Note:** Only supported in the master drive. |
| 2 | Read Remote:<br>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).<br>The node number of a drive is defined by parameter *57.03*.<br>**Note:** Only supported in the master drive. |
| 3 | Follower P2P:<br>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).<br>The node number of a drive is defined by parameter *57.03*.<br>**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*. |
| 4 | Standard Multicast:<br>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).<br>Which multicast group a drive belongs to is defined by the Std Mcast Group input of the *D2D_Conf* block.<br>A token from the master drive is required for a follower to be able to send the message. See the block *D2D_McastToken*. |
| 5 | Broadcast:<br>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.<br>A token from the master drive is required for a follower to be able to send the message. See block *D2D_McastToken*.<br>**Note:** With this message type, the Target Node/Grp input must be connected in DriveSPC even if not used. |

The Target Node/Grp input specifies the target drive or multicast group of drives depending on message type. See the message type explanations above.

**Note:** The input must be connected in DriveSPC even if not used.

The LocalDsNr input specifies the number of the local dataset used as the source or the target of the message.

The RemoteDsNr input specifies the number of the remote dataset used as the target or the source of the message.

The Sent msg count output is a wrap-around counter of successfully sent messages.

The error codes indicated by the Error output are as follows:

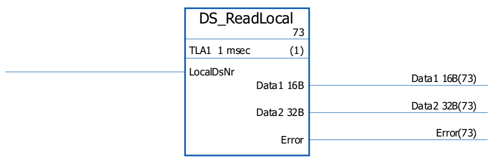
| Bit | Description |
|---|---|
| 0 | D2D_MODE_ERR: Drive-to-drive communication not activated, or message type not supported in current drive-to-drive mode (master/follower) |
| 1 | LOCAL_DS_ERR: LocalDsNr input out of range (16…199) |
| 2 | TARGET_NODE_ERR: Target Node/Grp input out of range (1…62) |
| 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** | Message type (Msg Type): INT<br>Target node or multicast group (Target Node/Grp): INT<br>Local dataset number (LocalDsNr): INT<br>Remote dataset number (RemoteDsNr): INT |
| **Outputs** | Successfully sent messages counter (Sent msg count): DINT<br>Error output (Error): PB |

## DS_Read_Local

(10094)

**Illustration**




**Execution time**

-

**Operation**

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.

The LocalDsNr input defines the number of the dataset to be read.

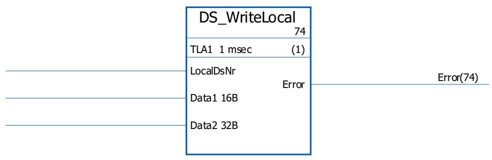
The error codes indicated by the Error output are as follows:

| Bit | Description |
|---|---|
| 1 | LOCAL_DS_ERR: LocalDsNr out of range (16…199) |

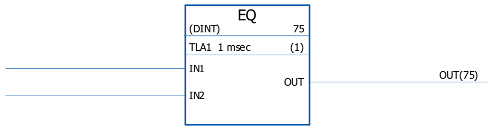
| | |
|---|---|
| **Inputs** | Local dataset number (LocalDsNr): INT |
| **Outputs** | Contents of dataset (Data1 16B): INT<br>Contents of dataset (Data2 32B): DINT<br>Error output (Error): DINT |

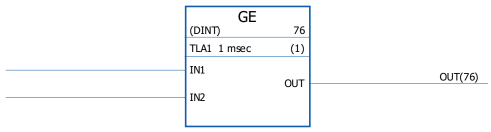
<table>
<tr><th colspan="2">

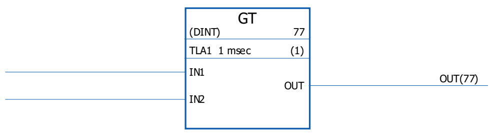
## DS_WriteLocal

(10093)
</th></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>-</td></tr>
<tr><td>Operation</td><td>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.<br>The error codes indicated by the Error output are as follows:
<table>
<tr><th>Bit</th><th>Description</th></tr>
<tr><td>1</td><td>LOCAL_DS_ERR: LocalDsNr out of range (16…199)</td></tr>
</table></td></tr>
<tr><td>Inputs</td><td>Local dataset number (LocalDsNr): INT<br>Contents of dataset (Data1 16B): INT<br>Contents of dataset (Data2 32B): DINT</td></tr>
<tr><td>Outputs</td><td>Error output (Error): DINT</td></tr>
</table>

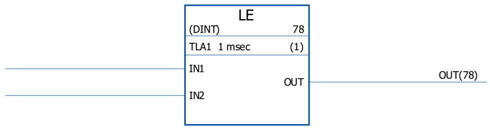
# Comparison

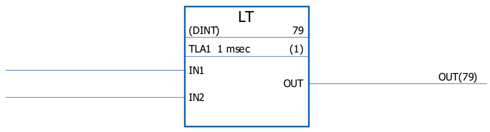
| **EQ**<br>(10040) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.89 μs (when two inputs are used) + 0.43 μs (for every additional input). When all inputs are used, the execution time is 13.87 μs. |
| **Operation** | The output (OUT) is 1 if all the connected input values are equal (IN1 = IN2 = … = IN32). Otherwise the output is 0. |
| **Inputs** | The input data type and the number of inputs (2…32) are selected by the user.<br>Input (IN1…IN32): INT, DINT, REAL, REAL24 |
| **Outputs** | Output (OUT): Boolean |

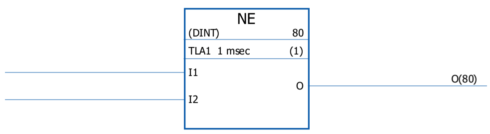
| **GE >=**<br>(10041) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.89 μs (when two inputs are used) + 0.43 μs (for every additional input). When all inputs are used, the execution time is 13.87 μs. |
| **Operation** | The output (OUT) is 1 if (IN1 > IN2) & (IN2 > IN3) & … & (IN31 > IN32). Otherwise the output is 0. |
| **Inputs** | The input data type and the number of inputs (2…32) are selected by the user.<br>Input (IN1…IN32): INT, DINT, REAL, REAL24 |
| **Outputs** | Output (OUT): Boolean |

| **GT >**<br>(10042) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.89 μs (when two inputs are used) + 0.43 μs (for every additional input). When all inputs are used, the execution time is 13.87 μs. |
| **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.<br>Input (IN1…IN32): INT, DINT, REAL, REAL24 |

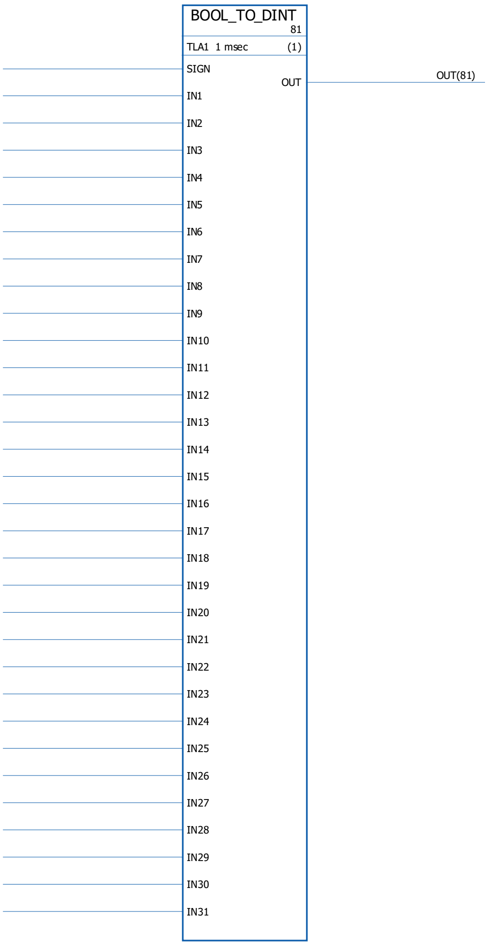
| | |
|---|---|
| **Outputs** | Output (OUT): Boolean |

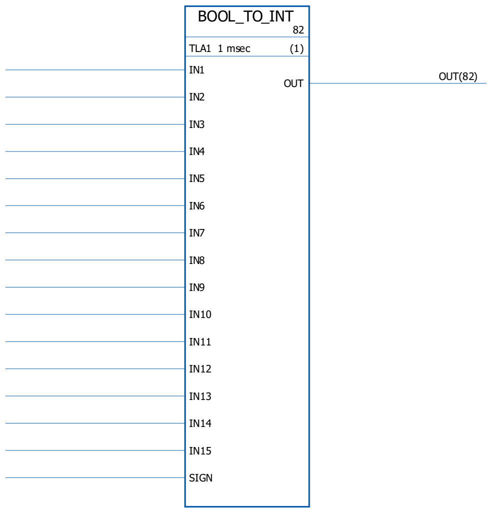
| **LE <=**<br>(10043) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.89 µs (when two inputs are used) + 0.43 µs (for every additional input). When all inputs are used, the execution time is 13.87 µs. |
| **Operation** | Output (OUT) is 1 if (IN1 $\leq$ IN2) & (IN2 $\leq$ IN3) & … & (IN31 $\leq$ IN32). Otherwise the output is 0. |
| **Inputs** | The input data type and the number of inputs (2…32) are selected by the user.<br>Input (IN1…IN32): INT, DINT, REAL, REAL24 |
| **Outputs** | Output (OUT): Boolean |

| **LT <**<br>(10044) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.89 µs (when two inputs are used) + 0.43 µs (for every additional input). When all inputs are used, the execution time is 13.87 µs. |
| **Operation** | 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.<br>Input (IN1…IN32): INT, DINT, REAL, REAL24 |
| **Outputs** | Output (OUT): Boolean |

| **NE <>**<br>(10045) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.44 µs |
| **Operation** | The output (O) is 1 if I1 <> I2. Otherwise the output is 0. |
| **Inputs** | The input data type is selected by the user.<br>Input (I1, I2): INT, DINT, REAL, REAL24 |
| **Outputs** | Output (O): Boolean |

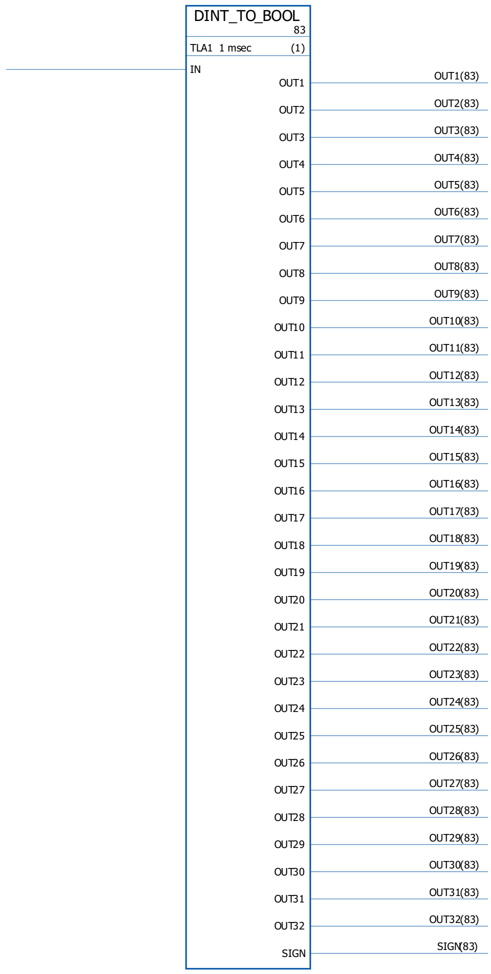
# Conversion

| **BOOL_TO_DINT**<br>(10018) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 13.47 µs |
| **Operation** | 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.<br>Example:<br>IN1 = 1, IN2 = 0, IN3…IN31 = 1, SIGN = 1<br>OUT = 1111 1111 1111 1111 1111 1111 1111 1101<br>SIGN IN31,Ä¶I |
| **Inputs** | Sign input (SIGN): Boolean<br>Input (IN1…IN31): Boolean |
| **Outputs** | Output (OUT): DINT (31 bits + sign) |

| **BOOL_TO_INT**<br><br>(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.<br>Example:<br>IN1…IN15 = 1, SIGN = 0<br>OUT = $\underbrace{0}_{\text{SIGN}}\underbrace{111\ 1111\ 1111\ 1111}_{\text{IN15,Ä¶I}}$ |
| **Inputs** | Input (IN1…IN15): Boolean<br>Sign input (SIGN): Boolean |
| **Outputs** | Output (OUT): DINT (15 bits + sign) |

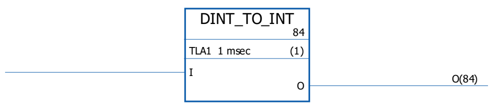
## DINT_TO_BOOL

(10020)

| | |
|---|---|
| **Illustration** |  |
| **Execution time** | 11.98 µs |
| **Operation** | The boolean output (OUT1…OUT32) values are formed from the 32-bit integer input (IN) value.<br>Example:<br>IN = $\underbrace{0}_{\text{SIGN}}$ $\underbrace{\text{111 1111 1111 1111 1111 1111 1111 1100}}_{\text{OUT32,Ä¶OU}}$ |
| **Inputs** | Input (IN): DINT |
| **Outputs** | Output (OUT1…OUT32): Boolean<br>Sign output (SIGN): Boolean |

## DINT_TO_INT

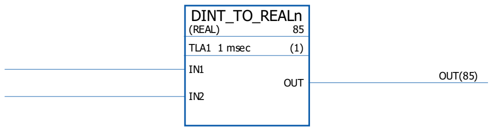
(10021)

| | |
|---|---|
| **Illustration** |  |

<table>
<tr><td>Execution time</td><td>0.53 µs</td></tr>
<tr><td>Operation</td><td>The output (O) value is a 16-bit integer value of the 32-bit integer input (I) value.<br>Examples:
<table>
<tr><td>I (31 bits + sign)</td><td>O (15 bits + sign)</td></tr>
<tr><td>2147483647</td><td>32767</td></tr>
<tr><td>-2147483648</td><td>-32767</td></tr>
<tr><td>0</td><td>0</td></tr>
</table>
</td></tr>
<tr><td>Inputs</td><td>Input (I): DINT</td></tr>
<tr><td>Outputs</td><td>Output (O): INT</td></tr>
</table>

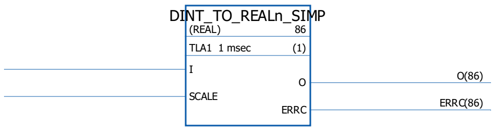
## DINT_TO_REALn

(10023)

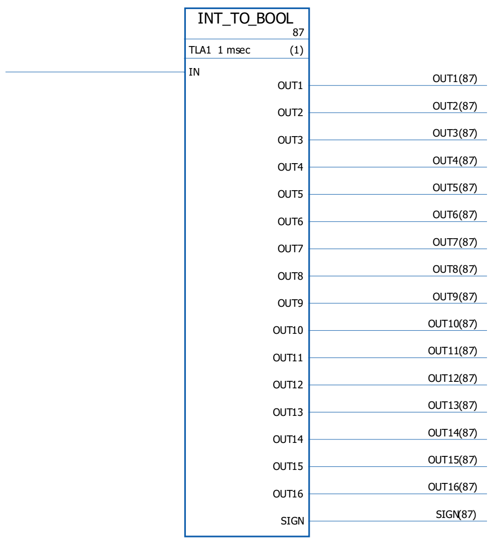
<table>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>7.25 µs</td></tr>
<tr><td>Operation</td><td>The output (OUT) is the REAL/REAL24 equivalent of the input (IN). Input IN1 is the integer value and input IN2 is the fractional value.<br>If one (or both) of the input values is negative, the output value is negative.<br>Example (from DINT to REAL):<br>When IN1 = 2 and IN2 = 3276, OUT = 2.04999.<br>The output value is limited to the maximum value of the selected data type range.</td></tr>
<tr><td>Inputs</td><td>Input (IN1, IN2): DINT</td></tr>
<tr><td>Outputs</td><td>The output data type is selected by the user.<br>Output (OUT): REAL, REAL24</td></tr>
</table>

## DINT_TO_REALn_SIMP

(10022)

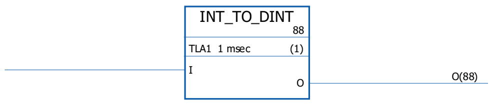
<table>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>6.53 µs</td></tr>
</table>

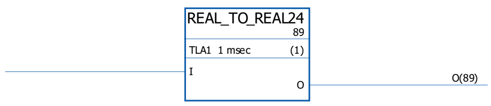
<table>
<tr><td>Operation</td><td>The output (O) is the REAL/REAL24 equivalent of the input (I) divided by the scale input (SCALE).<br>Error codes indicated at the error output (ERRC) are as follows:
<table>
<tr><th>Error code</th><th>Description</th></tr>
<tr><td>0</td><td>No error</td></tr>
<tr><td>1001</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>
</table>
Example (from DINT to REAL24):<br>When I = 205 and SCALE = 100, I/SCALE = 205 /100 = 2.05 and O = 2.04999.</td></tr>
<tr><td>Inputs</td><td>Input (I): DINT<br>Scale input (SCALE): DINT</td></tr>
<tr><td>Outputs</td><td>The output data type is selected by the user.<br>Output (O): REAL, REAL24<br>Error output (ERRC): DINT</td></tr>
</table>

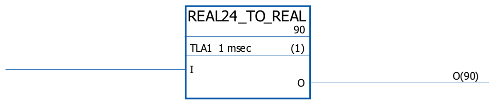
<table>
<tr><th colspan="2">INT_TO_BOOL<br>(10024)</th></tr>
<tr><td>Illustration</td><td>


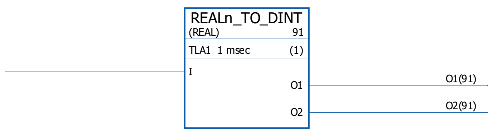
</td></tr>
<tr><td>Execution time</td><td>4.31 µs</td></tr>
<tr><td>Operation</td><td>The boolean output (OUT1…OUT16) values are formed from the 16-bit integer input (IN) value.<br>Example:<br>IN = 0111 1111 1111 1111<br>SIGN OUT16,Ä¶OU</td></tr>
</table>

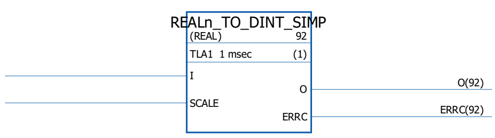
<table>
<tr><td>Inputs</td><td>Input (IN): INT</td></tr>
<tr><td>Outputs</td><td>Output (OUT1…OUT16): Boolean<br>Sign output (SIGN): Boolean</td></tr>
</table>

<table>
<tr><td colspan="2">INT_TO_DINT<br>(10025)</td></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>0.33 µs</td></tr>
<tr><td>Operation</td><td>The output (O) value is a 32-bit integer value of the 16-bit integer input (I) value.
<table>
<tr><th>I</th><th>O</th></tr>
<tr><td>32767</td><td>32767</td></tr>
<tr><td>-32767</td><td>-32767</td></tr>
<tr><td>0</td><td>0</td></tr>
</table></td></tr>
<tr><td>Inputs</td><td>Input (I): INT</td></tr>
<tr><td>Outputs</td><td>Output (O): DINT</td></tr>
</table>

<table>
<tr><td colspan="2">REAL_TO_REAL24<br>(10026)</td></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>1.35 µs</td></tr>
<tr><td>Operation</td><td>Output (O) is the REAL24 equivalent of the REAL input (I).<br>The output value is limited to the maximum value of the data type.<br>Example:<br>$I = \underbrace{0000\ 0000\ 0010\ 0110}_{\text{Integer value}}\ \underbrace{1111\ 1111\ 1111\ 1111}_{\text{Fractional value}}$<br>$O = \underbrace{0010\ 0110}_{\text{Integer value}}\ \underbrace{1111\ 1111\ 1111\ 1111\ 0000\ 0000}_{\text{Fractional value}}$</td></tr>
<tr><td>Inputs</td><td>Input (I): REAL</td></tr>
<tr><td>Outputs</td><td>Output (O): REAL24</td></tr>
</table>

| **REAL24_TO_REAL** (10027) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 1.20 µs |
| **Operation** | Output (O) is the REAL equivalent of the REAL24 input (I).<br>The output value is limited to the maximum value of the data type range.<br>Example:<br>$I = \underbrace{0010\ 0110}_{\text{Integer value}}\ \underbrace{1111\ 1111\ 1111\ 1111\ 0000\ 0000}_{\text{Fractional value}}$<br>$O = \underbrace{0000\ 0000\ 0010\ 0110}_{\text{Integer value}}\ \underbrace{1111\ 1111\ 1111\ 1111}_{\text{Fractional value}}$ |
| **Inputs** | Input (I): REAL24 |
| **Outputs** | Output (O): REAL |

| **REALn_TO_DINT** (10029) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 6.45 µs |
| **Operation** | 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.<br>The output value is limited to the maximum value of the data type range.<br>Example (from REAL to DINT):<br>When I = 2.04998779297, O1 = 2 and O2 = 3276. |
| **Inputs** | The input data type is selected by the user.<br>Input (I): REAL, REAL24 |
| **Outputs** | Output (O1, O2): DINT |

| **REALn_TO_DINT_SIMP** (10028) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 5.54 µs |

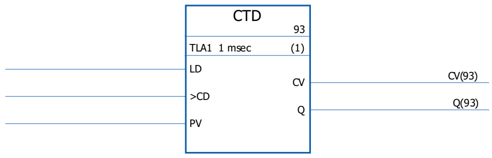
<table>
<tr><td>Operation</td><td>Output (O) is the 32-bit integer equivalent of the REAL/REAL24 input (I) multiplied by the scale input (SCALE).<br>Error codes are indicated by the error output (ERRC) as follows:
<table>
<tr><th>Error code</th><th>Description</th></tr>
<tr><td>0</td><td>No error</td></tr>
<tr><td>1001</td><td>The calculated integer value exceeds the minimum value. The output is set to the minimum value.</td></tr>
<tr><td>1002</td><td>The calculated integer value exceeds the maximum value. The output is set to the maximum value.</td></tr>
<tr><td>1003</td><td>Scale input is 0. The output is set to 0.</td></tr>
<tr><td>1004</td><td>Incorrect scale input, i.e. scale input is < 0 or is not a factor of 10.</td></tr>
</table>
Example (from REAL to DINT):<br>When I = 2.04998779297and SCALE = 100, O = 204.</td></tr>
<tr><td>Inputs</td><td>The input data type is selected by the user.<br>Input (I): REAL, REAL24<br>Scale input (SCALE): DINT</td></tr>
<tr><td>Outputs</td><td>Output (O): DINT<br>Error output (ERRC): DINT</td></tr>
</table>

# Counters

## CTD

(10047)

**Illustration**




**Execution time**

0.92 µs

**Operation**

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.

The status output (Q) is 1 if the counter output (CV) value $\leq$ 0.

Example:

| 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 |

$CV_{prev}$ is the previous cycle counter output value.

**Inputs**

Load input (LD): Boolean

Counter input (CD): Boolean

Preset input (PV): INT

**Outputs**

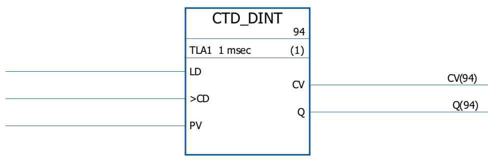
Counter output (CV): INT

Status output (Q): Boolean

## CTD_DINT

(10046)

**Illustration**




**Execution time**

0.92 µs

**Operation**

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.

The status output (Q) is 1 if the counter output (CV) value < 0.

Example:

| **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 |

$CV_{prev}$ is the previous cycle counter output value.

**Inputs**

Load input (LD): Boolean

Counter input (CD): Boolean

Preset input (PV): DINT

**Outputs**

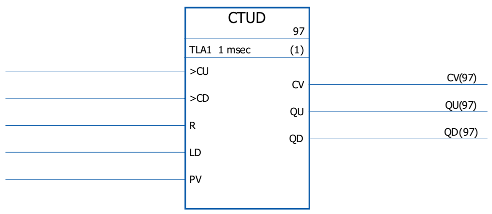
Counter output (CV): DINT

Status output (Q): Boolean

## CTU

(10049)

**Illustration**




**Execution time**

0.92 µs

**Operation**

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.

The counter output (CV) is reset to 0 if the reset input (R) is 1.

The status output (Q) is 1 if the counter output (CV) value ≥ preset input (PV) value.

Example:

| **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 |
| $CV_{prev}$ is the previous cycle counter output value. | | | | | |

**Inputs**

Counter input (CU): Boolean

Reset input (R): Boolean

Preset input (PV): INT

**Outputs**

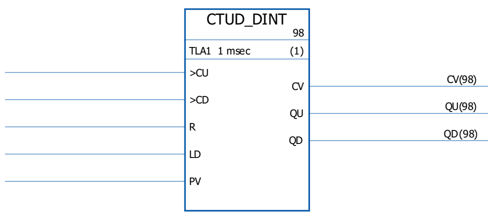
Counter output (CV): INT

Status output (Q): Boolean

## CTU_DINT

(10048)

**Illustration**




**Execution time**

0.92 µs

**Operation**

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.

The counter output (CV) is reset to 0 if the reset input (R) is 1.

The status output (Q) is 1 if the counter output (CV) value ≥ preset input (PV) value.

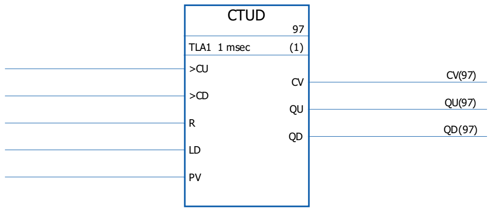
Example:

| **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 |
| $CV_{prev}$ is the previous cycle counter output value. | | | | | |

| | |
|---|---|
| **Inputs** | Counter input (CU): Boolean<br>Reset input (R): Boolean<br>Preset input (PV): DINT |
| **Outputs** | Counter output (CV): DINT<br>Status output (Q): Boolean |

## CTUD

(10051)

| | |
|---|---|
| **Illustration** |  |
| **Execution time** | 1.40 µs |
| **Operation** | 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.<br>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.<br>If the load input (LD) is 1, the preset input (PV) value is stored as the counter output (CV) value.<br>The counter output (CV) is reset to 0 if the reset input (R) is 1.<br>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.<br>The up counter status output (QU) is 1 if the counter output (CV) value $\geq$ preset input (PV) value.<br>The down counter status output (QD) is 1 if the counter output (CV) value $\leq$ 0. |

Example:

| 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 |

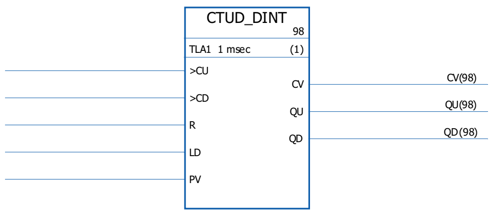
$CV_{prev}$ is the previous cycle counter output value.

| | |
|---|---|
| **Inputs** | Up counter input (CU): Boolean<br>Down counter input (CD): Boolean<br>Reset input (R): Boolean<br>Load input (LD): Boolean<br>Preset input (PV): INT |
| **Outputs** | Counter output (CV): INT<br>Up counter status output (QU): Boolean<br>Down counter status output (QD): Boolean |

## CTUD_DINT

(10051)

**Illustration**




| | |
|---|---|
| **Execution time** | 1.40 µs |

**Operation**

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.

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.

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.

If the load input (LD) value is 1, the preset input (PV) value is stored as the counter output (CV) value.

The counter output (CV) is reset to 0 if the reset input (R) is 1.

The up counter status output (QU) is 1 if the counter output (CV) value $\geq$ preset input (PV) value.

The down counter status output (QD) is 1 if the counter output (CV) value $\leq$ 0.

Example:

| 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 |

$CV_{prev}$ is the previous cycle counter output value.

**Inputs**

Up counter input (CU): Boolean

Down counter input (CD): Boolean

Reset input (R): Boolean

Load input (LD): Boolean

Preset input (PV): DINT

**Outputs**

Counter output (CV): DINT

Up counter status output (QU): Boolean

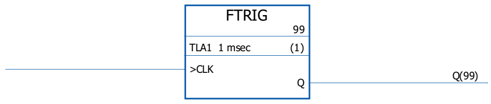
Down counter status output (QD): Boolean

# Edge & bistable

## FTRIG

(10030)

**Illustration**




**Execution time**

0.38 µs

**Operation**

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.

| $\mathbf{CLK_{previous}}$ | **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 |
| $CLK_{previous}$ is the previous cycle output value. | | |

**Inputs**

Clock input (CLK): Boolean

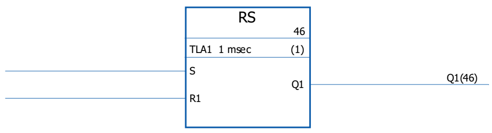
**Outputs**

Output (Q): Boolean

## RS

(10032)

**Illustration**




**Execution time**

0.38 µs

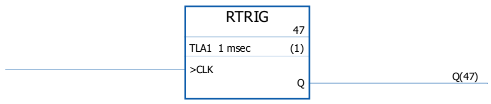
**Operation**

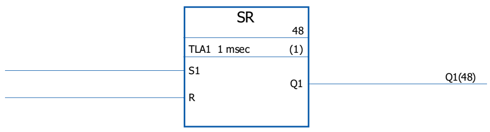
The output (Q1) is 1 if the set input (S) is 1 and the reset input (R1) is 0. The output will retain the previous output state if the set input (S) and the reset input (R1) are 0. The output is 0 if the reset input is 1.

Truth table:

| **S** | **R1** | $\mathbf{Q1_{previous}}$ | **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 |
| $Q_{previous}$ is the previous cycle output value. | | | |

| | |
|---|---|
| **Inputs** | Set input (S): Boolean<br>Reset input (R1): Boolean |
| **Outputs** | Output (Q1): Boolean |

<table>
<tr><td colspan="2">RTRIG<br>(10031)</td></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>0.38 µs</td></tr>
<tr><td>Operation</td><td>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.
<table>
<tr><th>CLK<sub>previous</sub></th><th>CLK</th><th>Q</th></tr>
<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>
<tr><td colspan="3">$CLK_{previous}$ is the previous cycle output value.</td></tr>
</table>
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.</td></tr>
<tr><td>Inputs</td><td>Clock input (CLK): Boolean</td></tr>
<tr><td>Outputs</td><td>Output (Q): Boolean</td></tr>
</table>

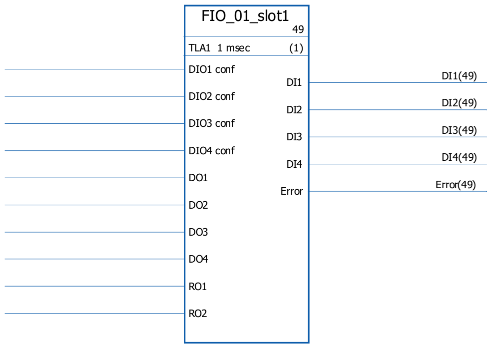
<table>
<tr><td colspan="2">SR<br>(10033)</td></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>0.38 µs</td></tr>
</table>

<table>
<tr><td>Operation</td><td>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.<br>Truth table:
<table>
<tr><th>S1</th><th>R</th><th>$Q1_{previous}$</th><th>Q1</th></tr>
<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>
<tr><td colspan="4">$Q1_{previous}$ is the previous cycle output value.</td></tr>
</table>
</td></tr>
<tr><td>Inputs</td><td>Set input (S1): Boolean<br>Reset input (R): Boolean</td></tr>
<tr><td>Outputs</td><td>Output (Q1): Boolean</td></tr>
</table>

# Extensions (ACS850 only)

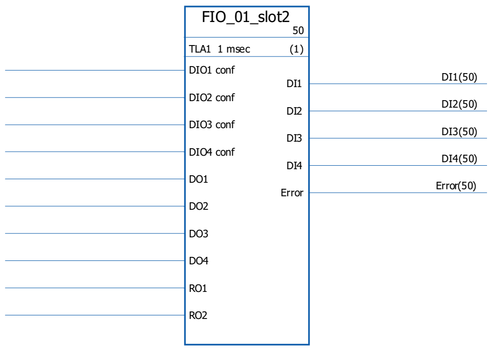
## FIO_01_slot1

(10084)

| | |
|---|---|
| **Illustration** |   |
| **Execution time** | 8.6 µs |
| **Operation** | 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.<br>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.<br>The RO1 and RO2 inputs define the state of the relay outputs of the FIO-01 (0 = not energized, 1 = energized).<br>The DIx outputs show the state of the DIOs. |
| **Inputs** | Digital input/output mode selection (DIO1 conf … DIO4 conf): Boolean<br>Digital output state selection (DO1…DO4): Boolean<br>Relay output state selection (RO1, RO2): Boolean |
| **Outputs** | Digital input/output state (DI1…DI4): Boolean<br>Error output (Error): DINT (0 = No error; 1 = Application program memory full) |

## FIO_01_slot2

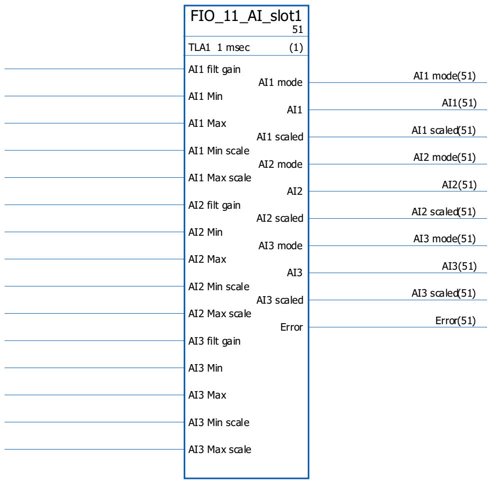
(10085)

| | |
|---|---|
| **Illustration** |   |
| **Execution time** | 8.6 µs |

| | |
|---|---|
| **Operation** | 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.<br>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.<br>The RO1 and RO2 inputs define the state of the relay outputs of the FIO-01 (0 = not energised, 1 = energised).<br>The DIx outputs show the state of the DIOs. |
| **Inputs** | Digital input/output mode selection (DIO1 conf … DIO4 conf): Boolean<br>Digital output state selection (DO1…DO4): Boolean<br>Relay output state selection (RO1, RO2): Boolean |
| **Outputs** | Digital input/output state (DI1…DI4): Boolean<br>Error output (Error): DINT (0 = No error; 1 = Application program memory full) |

## FIO_11_AI_slot1

(10088)

| | |
|---|---|
| **Illustration** |  |
| **Execution time** | 11.1 µs |

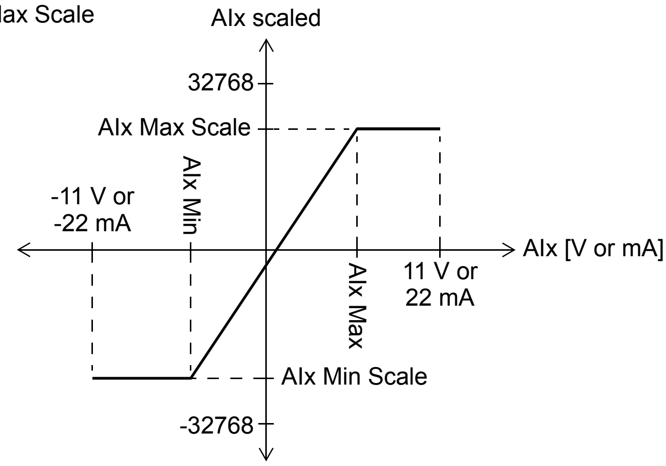
**Operation**

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.

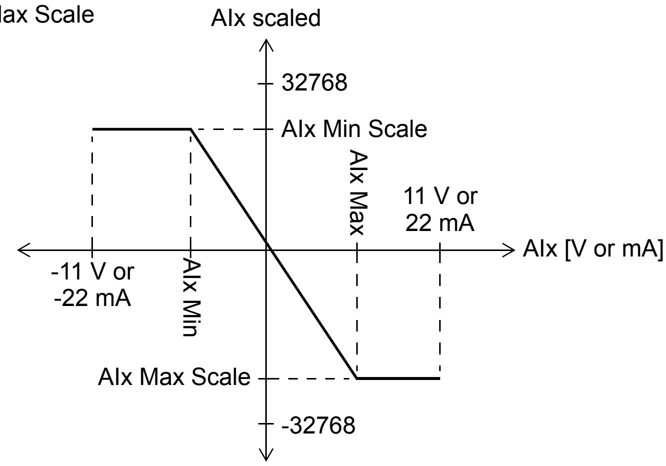
The block outputs both the unscaled (AIx) and scaled (AIx scaled) actual values of each analogue input. The scaling is based on the relationship between the ranges AIx min … AIx max and AIx min scale … AIx max scale.

AIx Min must be smaller than AIx Max; AIx Max Scale can be greater or smaller than AIx Min Scale.

AIx Min Scale < AIx Max Scale




AIx Min Scale > AIx Max Scale




The AIx filt gain inputs determine a filtering time for each input as follows:

| AIx filt gain | Filtering time | Notes |
|---|---|---|
| 0 | No filtering | |
| 1 | 125 µs | Recommended setting |
| 2 | 250 µs | |
| 3 | 500 µs | |
| 4 | 1 ms | |
| 5 | 2 ms | |
| 6 | 4 ms | |
| 7 | 7.9375 ms | |

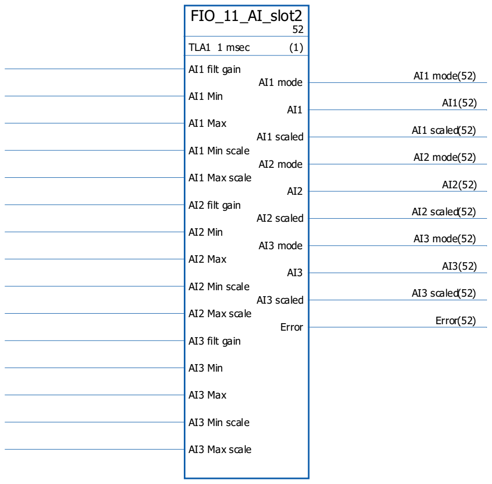
The AIx mode outputs show whether the corresponding input is voltage (0) or current (1). The voltage/current selection is made using the hardware switches on the FIO-11.

| | |
|---|---|
| **Inputs** | Analogue input filter gain selection (AI1 filt gain … AI3 filt gain): INT<br>Minimum value of input signal (AI1 Min … AI3 Min): REAL (≥ -11 V or -22 mA)<br>Maximum value of input signal (AI1 Max … AI3 Max): REAL (≤ 11 V or 22 mA)<br>Minimum value of scaled output signal (AI1 Min scale … AI3 Min scale): REAL<br>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<br>Value of analogue input (AI1 … AI3): REAL<br>Scaled value of analogue input (AI1 scaled … AI3 scaled): REAL<br>Error output (Error): DINT (0 = No error; 1 = Application program memory full) |

## FIO_11_AI_slot2

(10089)

**Illustration**




**Execution time**

11.1 µs

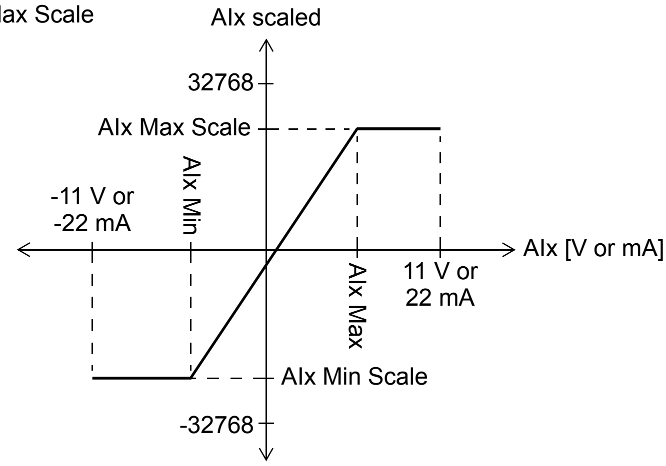
**Operation**

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.

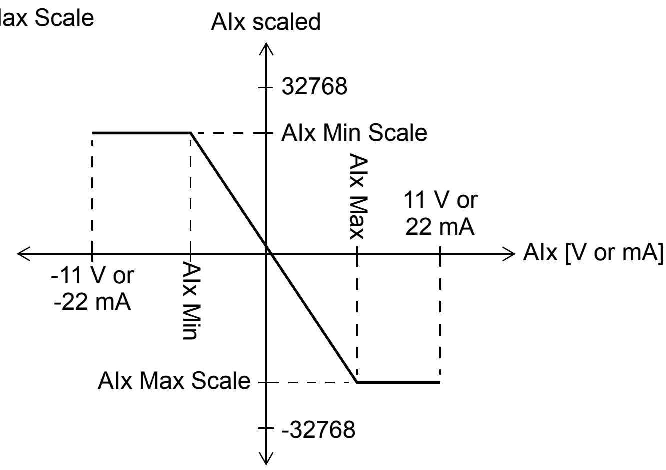
The block outputs both the unscaled (AIx) and scaled (AIx scaled) actual values of each analogue input. The scaling is based on the relationship between the ranges AIx min … AIx max and AIx min scale … AIx max scale.

AIx Min must be smaller than AIx Max; AIx Max Scale can be greater or smaller than AIx Min Scale.

AIx Min Scale < AIx Max Scale



AIx Min Scale > AIx Max Scale




The AIx filt gain inputs determine a filtering time for each input as follows:

| AIx filt gain | Filtering time | Notes |
|---|---|---|
| 0 | No filtering | |
| 1 | 125 µs | Recommended setting |
| 2 | 250 µs | |
| 3 | 500 µs | |
| 4 | 1 ms | |
| 5 | 2 ms | |
| 6 | 4 ms | |
| 7 | 7.9375 ms | |

The AIx 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

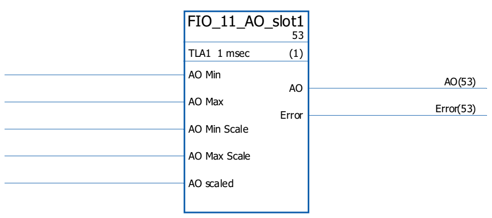
**Outputs**

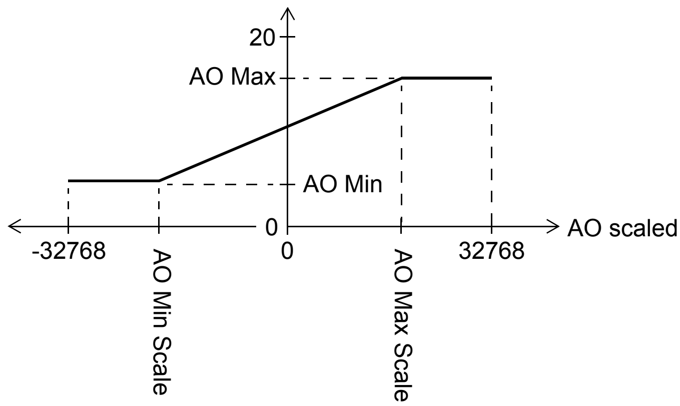
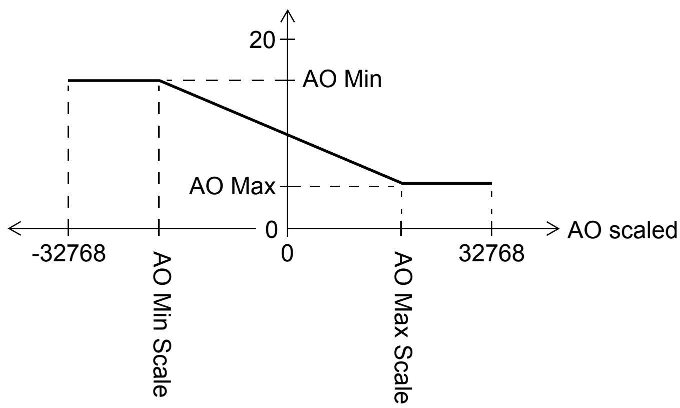
Analogue input mode (voltage or current) (AI1 mode … AI3 mode): Boolean
Value of analogue input (AI1 … AI3): REAL
Scaled value of analogue input (AI1 scaled … AI3 scaled): REAL
Error output (Error): DINT (0 = No error; 1 = Application program memory full)

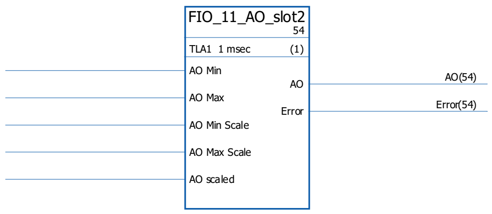
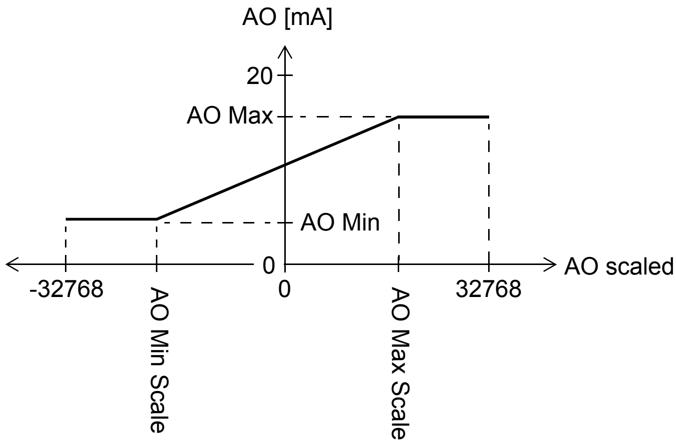
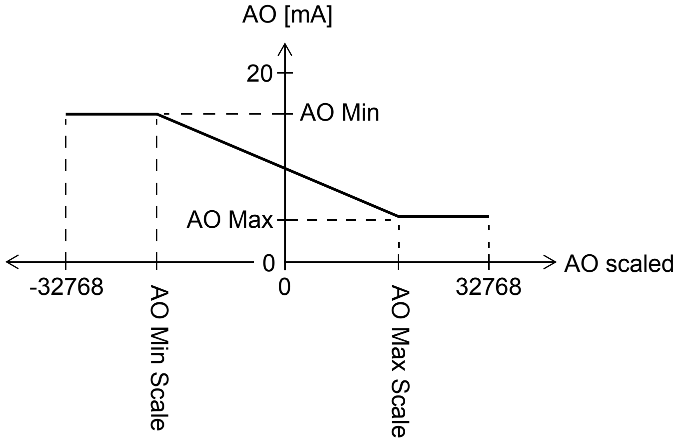
## FIO_11_AO_slot1

(10090)

**Illustration**



| | |
|---|---|
| **Execution time** | 4.9 µs |
| **Operation** | The block controls the analogue output (AO1) of a FIO-11 Analog I/O Extension mounted on slot 1 of the drive control unit.<br>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.<br>AO Min Scale must be smaller than AO Max Scale; AO Max can be greater or smaller than AO Min.<br>AO Min < AO Max<br><br>AO Min > AO Max<br> |
| **Inputs** | Minimum current signal (AO Min): REAL (0…20 mA)<br>Maximum current signal (AO Max): REAL (0…20 mA)<br>Minimum input signal (AO Min Scale): REAL<br>Maximum input signal (AO Max Scale): REAL<br>Input signal (AO scaled): REAL |
| **Outputs** | Analogue output current value (AO): REAL<br>Error output (Error): DINT (0 = No error; 1 = Application program memory full) |

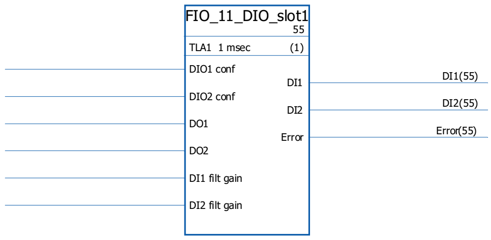
| **FIO_11_AO_slot2**<br>(10091) | |
|---|---|
| **Illustration** | <br> |
| **Execution time** | 4.9 µs |
| **Operation** | The block controls the analogue output (AO1) of a FIO-11 Analog I/O Extension mounted on slot 2 of the drive control unit.<br>The block converts the input signal (AO scaled) to 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.<br>AO Min Scale must be smaller than AO Max Scale; AO Max can be greater or smaller than AO Min.<br>AO Min < AO Max<br><br><br>AO Min > AO Max<br><br> |

| | |
|---|---|
| **Inputs** | Minimum current signal (AO Min): REAL (0…20 mA)<br>Maximum current signal (AO Max): REAL (0…20 mA)<br>Minimum input signal (AO Min Scale): REAL<br>Maximum input signal (AO Max Scale): REAL<br>Input signal (AO scaled): REAL |
| **Outputs** | Analogue output current value (AO): REAL<br>Error output (Error): DINT (0 = No error; 1 = Application program memory full) |

## FIO_11_DIO_slot1

(10086)

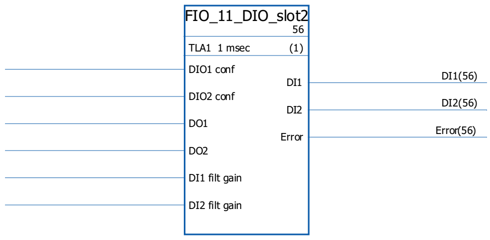
**Illustration**



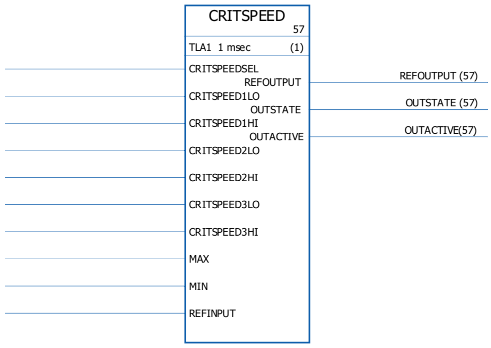
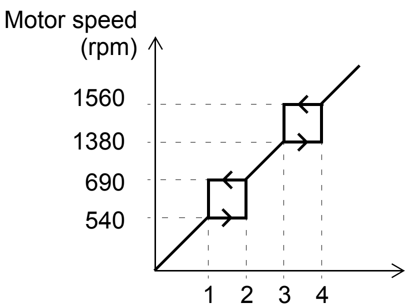

| | |
|---|---|
| **Execution time** | 6.0 µs |
| **Operation** | The block controls the two digital inputs/outputs (DIO1, DIO2) of a FIO-11 Digital I/O Extension mounted on slot 1 of the drive control unit.<br>The state of a DIOx conf input of the block determines whether the corresponding DIO on the 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.<br>The DIx outputs show the state of the DIOs.<br>The DIx filt gain inputs determine a filtering time for each input as follows: |

| DIx filt gain | Filtering time |
|---|---|
| 0 | 7.5 µs |
| 1 | 195 µs |
| 2 | 780 µs |
| 3 | 4.680 ms |

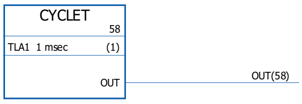
| | |
|---|---|
| **Inputs** | Digital input/output mode selection (DIO1 conf, DIO2 conf): Boolean<br>Digital output state selection (DO1, DO2): Boolean<br>Digital input filter gain selection (DI1 filt gain, DI2 filt gain): INT |
| **Outputs** | Digital input/output state (DI1, DI2): Boolean<br>Error output (Error): DINT (0 = No error; 1 = Application program memory full) |

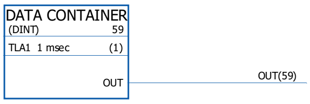
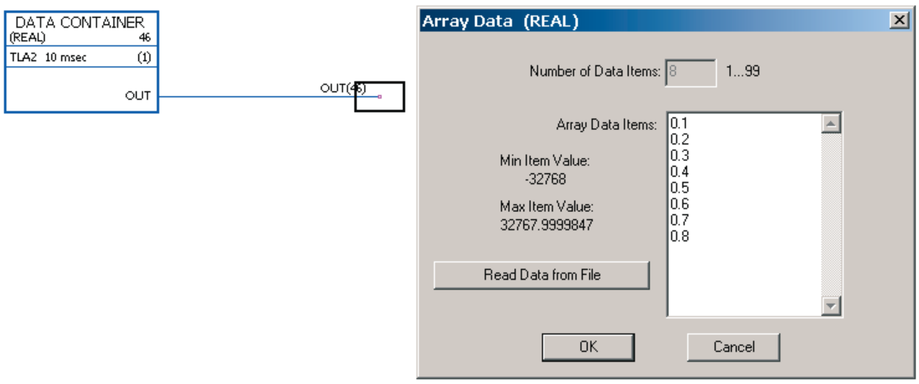
<table>
<tr><th colspan="2">FIO_11_DIO_slot2<br>(10087)</th></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>6.0 µs</td></tr>
<tr><td>Operation</td><td>The block controls the two digital inputs/outputs (DIO1, DIO2) of a FIO-11 Digital I/O Extension mounted on slot 2 of the drive control unit.<br>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.<br>The DIx outputs show the state of the DIOs.<br>The DIx filt gain inputs determine a filtering time for each input as follows:
<table>
<tr><th>DIx filt gain</th><th>Filtering time</th></tr>
<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>
</table></td></tr>
<tr><td>Inputs</td><td>Digital input/output mode selection (DIO1 conf, DIO2 conf): Boolean<br>Digital output state selection (DO1, DO2): Boolean<br>Digital input filter gain selection (DI1 filt gain, DI2 filt gain): INT</td></tr>
<tr><td>Outputs</td><td>Digital input/output state (DI1, DI2): Boolean<br>Error output (Error): DINT (0 = No error; 1 = Application program memory full)</td></tr>
</table>

# Feedback & algorithms

<table>
<tr><th colspan="2">CRITSPEED<br>(10068)</th></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>4.50 µs</td></tr>
<tr><td>Operation</td><td>
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.
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:
- activate the critical speeds function (CRITSPEEDSEL = 1),
- set the critical speed ranges as in the figure below.


<table>
<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>
Output OUTACTIVE is set to 1 when the output reference (REFOUTPUT) is different from the input reference (REFINPUT).
The output is limited by the defined minimum and maximum limits (MIN and MAX).
Output OUTSTATE indicates in which critical speed range the operation point is.
</td></tr>
<tr><td>Inputs</td><td>
Critical speed activation input (CRITSPEEDSEL): Boolean
Minimum/maximum critical speed range input (CRITSPEEDNLO / CRITSPEEDNHI): REAL
Maximum/minimum input (MAX/MIN): REAL
Reference input (REFINPUT): REAL
</td></tr>
<tr><td>Outputs</td><td>
Reference output (REFOUTPUT): REAL
Output state (OUTSTATE): REAL
Output active (OUTACTIVE): Boolean
</td></tr>
</table>

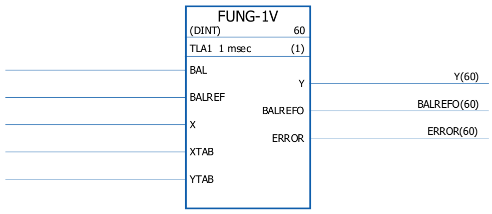
| **CYCLET**<br>(10074) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.00 µs |
| **Operation** | Output (OUT) is the time level of the CYCLET function block. |
| **Inputs** | - |
| **Outputs** | Output (OUT): DINT. 1 = 1 µs |

| **DATA CONTAINER**<br>(10073) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.00 µs |
| **Operation** | Output (OUT) is an array of data with values 1…99. The array can be used by the XTAB and YTAB tables in block *FUNG-1V*. 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.<br>Example:<br> |
| **Inputs** | - |
| **Outputs** | The output data type and the number of coordinate pairs are selected by the user.<br>Output (OUT): DINT, INT, REAL or REAL24 |

## FUNG-1V

(10072)

**Illustration**




**Execution time**

9.29 µs

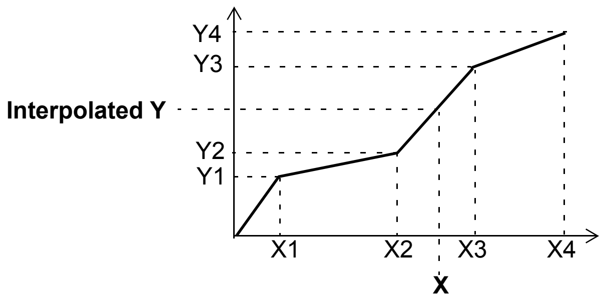
**Operation**

The output (Y) at the value of the input (X) is calculated with linear interpolation from a piecewise linear function.

$Y = Y_k + (X - X_k)(Y_{k+1} - Y_k) / (X_{k+1} - X_k)$

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).

XTAB and YTAB values are defined with the DriveSPC tool.




| X table (XTAB) | Y table (YTAB) |
|---|---|
| X1 | Y1 |
| X2 | Y2 |
| X3 | Y3 |
| … | … |
| X9 | Y9 |

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).

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.

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.

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.

**Inputs**

The input data type is selected by the user.

Balance input (BAL): Boolean

Balance reference input (BALREF): DINT, INT, REAL, REAL24.

X value input (X): DINT, INT, REAL, REAL24

X table input (XTAB): DINT, INT, REAL, REAL24

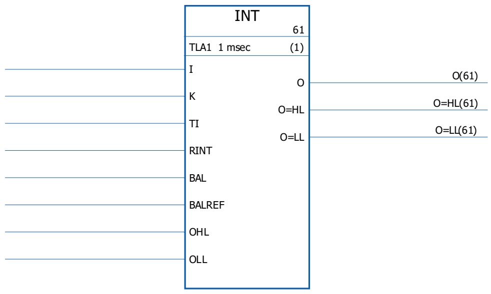
Y table input (YTAB): DINT, INT, REAL, REAL24

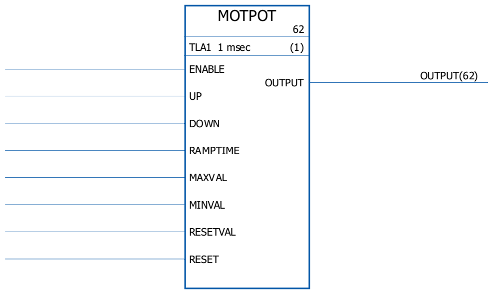
**Outputs**

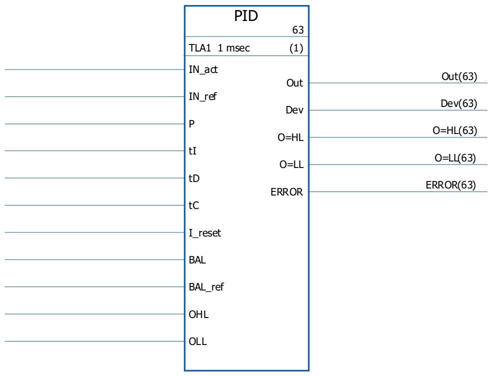
Y value output (Y): DINT, INT, REAL, REAL24

Balance reference output (BALREFO): DINT, INT, REAL, REAL24

Error output (ERROR): Boolean

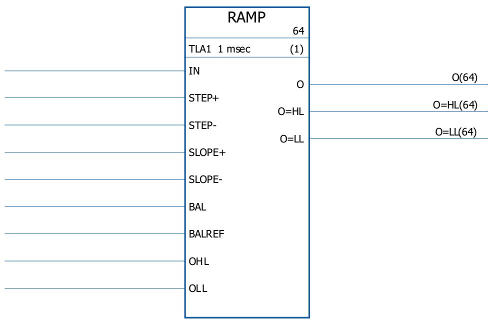
| **INT**<br>(10065) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 4.73 μs |
| **Operation** | The output (O) is the integrated value of the input (I):<br>$O(t) = K/TI (\int I(t)\, dt)$<br>Where TI is the integration time constant and K is the integration gain.<br>The step response for the integration is:<br>$O(t) = K \times I(t) \times t/TI$<br>The transfer function for the integration is:<br>$G(s) = K\ 1/sTI$<br>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.<br>The integration time constant is limited to value 2147483 ms. If the time constant is negative, zero time constant is used.<br>If the ratio between the cycle time and the integration time constant Ts/TI < 1, Ts/TI is set to 1.<br>The integrator is cleared when the reset input (RINT) is set to 1.<br>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. |
| **Inputs** | Input (I): REAL<br>Gain input (K): REAL<br>Integration time constant input (TI): DINT, 0…2147483 ms<br>Integrator reset input (RINT): Boolean<br>Balance input (BAL): Boolean<br>Balance reference input (BALREF): REAL<br>Output high limit input (OHL): REAL<br>Output low limit input (OLL): REAL |
| **Outputs** | Output (O): REAL<br>High limit output (O=HL): Boolean<br>Low limit output (O=LL): Boolean |

| **MOTPOT**<br>(10067) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 2.92 µs |
| **Operation** | The motor potentiometer function controls the rate of change of the output from the minimum to the maximum value and vice versa.<br>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.<br>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.<br>If the ENABLE input is 0, the output is zero.<br>Digital inputs are normally used as up and down inputs. |
| **Inputs** | Function enable input (ENABLE): Boolean<br>Up input (UP): Boolean<br>Down input (DOWN): Boolean<br>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)<br>Maximum reference input (MAXVAL): REAL<br>Minimum reference input (MINVAL): REAL<br>Reset value input (RESETVAL): REAL<br>Reset input (RESET): Boolean |
| **Outputs** | Output (OUTPUT) REAL |

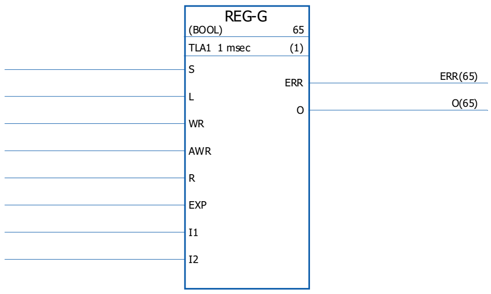
| **PID**<br>(10075) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 15.75 µs |
| **Operation** | The PID controller can be used for closed-loop control systems. The controller includes anti-windup correction and output limitation.<br>The PID controller output (Out) before limitation is the sum of the proportional ($U_P$), integral ($U_I$) and derivative ($U_D$) terms:<br>$Out_{unlimited}(t) = U_P(t) + U_I(t) + U_D(t)$<br>$U_P(t) = P \times Dev(t)$<br>$U_I(t) = P/tI \times [\int Dev(\tau)d\tau + tC \times (Out(t) - Out_{unlimited}(t))]$<br>$U_D(t) = P \times tD \times d(Dev(t))/dt$<br>Integrator:<br>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.<br>If integration time constant tI is 0, the integral term will not be updated.<br>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.<br>Limitation:<br>The output is limited by the defined minimum and maximum values, OLL and OHL:<br>If the actual value of the output reaches the specified minimum limit, output O=LL is set to 1.<br>If the actual value of the output reaches the specified maximum limit, output O=HL is set to 1.<br>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.<br>Error codes:<br>Error codes are indicated by the error output (ERROR) as follows |

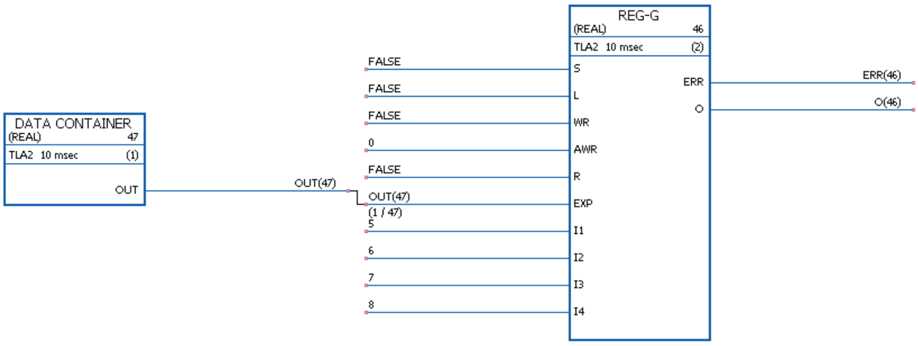
| Error code | Description |
|---|---|
| 1 | The minimum limit (OLL) exceeds the maximum limit (OHL). |
| 2 | Overflow with Up, Ui, or Ud calculation |

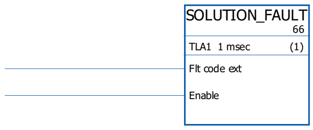
| | |
|---|---|
| | Balancing:<br>The balancing function (BAL) permits the output signal to track an external reference and gives a smooth return to the normal operation. If BAL is set to 1, the output (Out) is set to the value of the balance reference input (BAL_ref). Balance reference is limited by the defined minimum and maximum limits (OLL and OHL).<br>Anti-windup:<br>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. |
| **Inputs** | Actual input (IN_act): REAL<br>Reference input (IN_ref): REAL<br>Proportional gain input (P): REAL<br>Integration time constant input (tI): REAL. 1 = 1 ms<br>Derivation time constant input (tD): REAL. 1 = 1 ms<br>Antiwind-up correction time constant input (tC): IQ6. 1 = 1 ms<br>Integrator reset input (I_reset): Boolean<br>Balance input (BAL): Boolean<br>Balance reference input (BAL_ref): REAL<br>Output high limit input (OHL): REAL<br>Output low limit input (OLL): REAL |
| **Outputs** | Output (Out): REAL<br>Deviation output (Dev): REAL (= actual -reference = IN_act - IN_ref)<br>High limit output (O=HL): Boolean<br>Low limit output (O=LL): Boolean<br>Error code output (ERROR): INT32 |

| **RAMP**<br>(10066) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 4.23 µs |

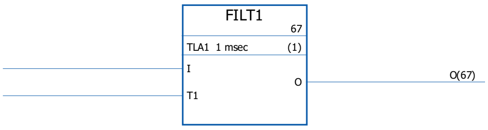
| | |
|---|---|
| **Operation** | Limits the rate of the change of the signal.<br>The input signal (IN) is connected directly to the output (O) if the input signal does not exceed the defined step change limits (STEP+ and STEP-). If the input signal change exceeds these limits, the output signal change 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.<br>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.<br>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). |
| **Inputs** | Input (IN): REAL<br>Maximum positive step change input (STEP+): REAL<br>Maximum negative step change input (STEP-): REAL<br>Ramp-up value per second input (SLOPE+): REAL<br>Ramp-down value per second input (SLOPE-): REAL<br>Balance input (BAL): Boolean<br>Balance reference input (BALREF): REAL<br>Output high limit input (OHL): REAL<br>Output low limit input (OLL): REAL |
| **Outputs** | Output (O): REAL<br>High limit output (O=HL): Boolean<br>Low limit output (O=LL): Boolean |

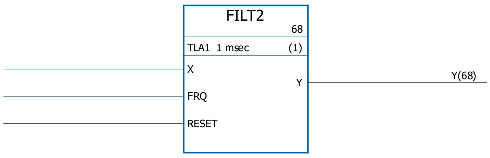
| **REG-G**<br>(10102) | |
|---|---|
| **Illustration** |  |
| **Execution time** | - |

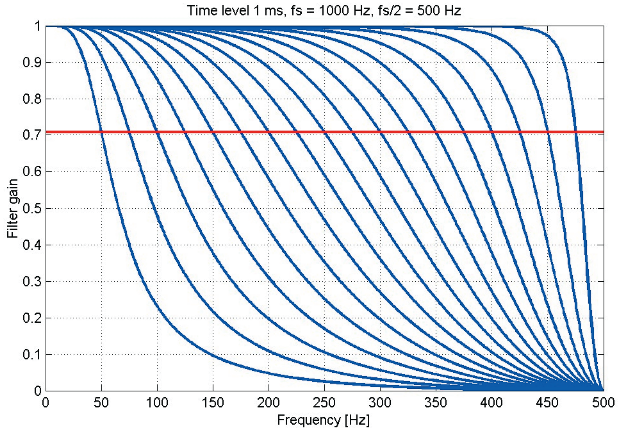
| | |
|---|---|
| **Operation** | Combines the array (group of variables) (if any) on the EXP input with the values of the I1…I32 pins to produce an output array. The data type of the arrays can be INT, DINT, REAL16, REAL24 or Boolean. The output array consists of the data from the EXP input and the values of the I1…In (in this order).<br>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.<br>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.<br>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.<br>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.<br>Whenever an error is detected, ERR is set within one cycle. No place in the register is affected when an error occurs.<br>Example:<br><br>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 [1,2,3,4,0,0,7,0]. |
| **Inputs** | Set (S): Boolean, INT, DINT, REAL, REAL24<br>Load (L): Boolean, INT, DINT, REAL, REAL24<br>Write (WR): Boolean, INT, DINT, REAL, REAL24<br>Write address (AWR): INT<br>Reset (R): Boolean<br>Expander (EXP): IArray<br>Data input (I1…I32): Boolean, INT, DINT, REAL, REAL24 |
| **Outputs** | Error (ERR): INT<br>Array data output (O): OC1 |

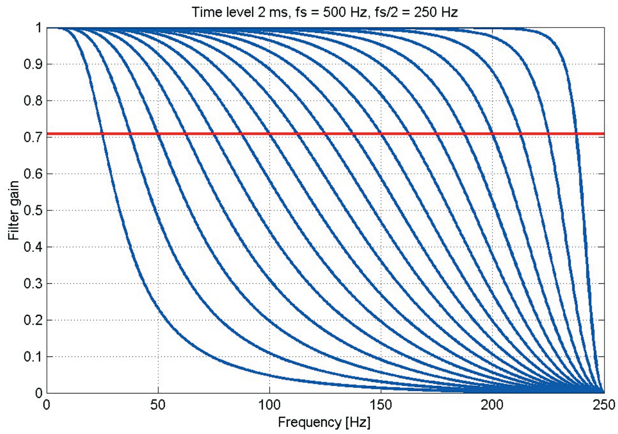
| SOLUTION_FAULT<br>(10097) | |
|---|---|
| **Illustration** |  |
| **Execution time** | - |
| **Operation** | When the block is enabled (by setting the Enable input to 1), a fault (F-0317 SOLUTION FAULT) is generated by the drive. The value of the Flt code ext input is recorded by the fault logger. |
| **Inputs** | Fault code extension (Flt code ext): DINT<br>Generate fault (Enable): Boolean |
| **Outputs** | - |

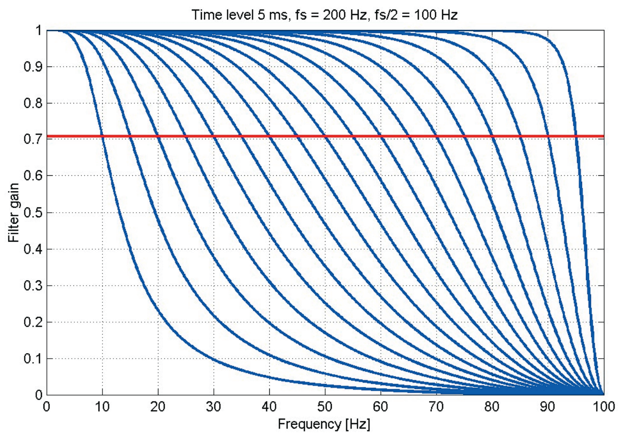
# Filters

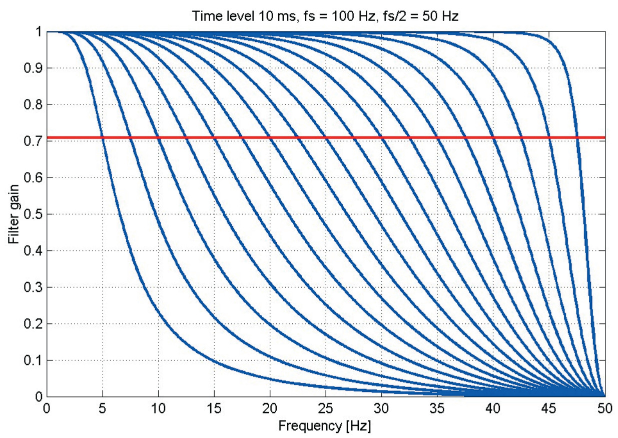
| **FILT1**<br>(10069) | |
|---|---|
| **Illustration** | <br> |
| **Execution time** | 7.59 µs |
| **Operation** | The output (O) is the filtered value of the input (I) value and the previous output value ($O_{prev}$). The FILT1 block acts as 1st order low pass filter.<br>**Note:** Filter time constant (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.<br>If T1 < Ts, the output value is the input value.<br>The step response for a single pole low pass filter is:<br>$O(t) = I(t) \times (1 - e^{-t/T1})$<br>The transfer function for a single pole low pass filter is:<br>$G(s) = 1/(1 + sT1)$ |
| **Inputs** | Input (I): REAL<br>Filter time constant input (T1): DINT, 1 = 1 ms |
| **Outputs** | Output (O): REAL |

| **FILT2**<br>(10070) | |
|---|---|
| **Illustration** | <br> |
| **Execution time** | 6.30 µs |
| **Operation** | The output (Y) is the filtered value of the input (X). The FILT2 block acts as a 2nd order low pass filter.<br>When the RESET input value is set to 1, the input is connected to the output without filtering.<br>**Notes:**<br>• The -3 dB cutoff frequency (FRQ) is limited to its maximum value (16383 Hz).<br>• 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.<br>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. |

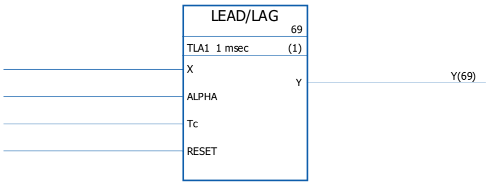

Time level 1 ms, fs = 1000 Hz, fs/2 = 500 Hz
Filter gain
Frequency [Hz]


Time level 2 ms, fs = 500 Hz, fs/2 = 250 Hz
Filter gain
Frequency [Hz]

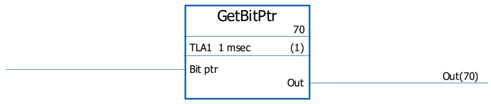

Time level 5 ms, fs = 200 Hz, fs/2 = 100 Hz
Filter gain
Frequency [Hz]

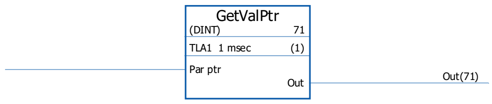

Time level 10 ms, fs = 100 Hz, fs/2 = 50 Hz
Filter gain
Frequency [Hz]

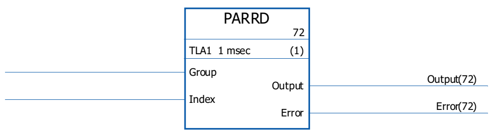
| | |
|---|---|
| **Inputs** | Input (X): REAL<br>-3 dB cutoff frequency input (FRQ): DINT (0…16383 Hz)<br>Reset input (RESET): Boolean |
| **Outputs** | Output (Y): REAL |

| **LEAD/LAG**<br>(10071) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 5.55 µs |
| **Operation** | 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.<br>The transfer function for a lead/lag filter is:<br>$(1 + \mathrm{ALPHA}T_c s) / (1 + T_c s)$<br>When RESET input is 1, the input value (X) is connected to the output (Y).<br>If ALPHA or Tc < 0, the negative input value is set to zero before filtering. |
| **Inputs** | Input (X): REAL<br>Lead/Lag filter type input (ALPHA): REAL<br>Time constant input (Tc): REAL<br>Reset input (RESET): Boolean |
| **Outputs** | Output (Y): REAL |

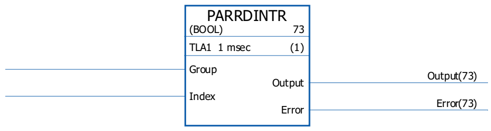
# Parameters

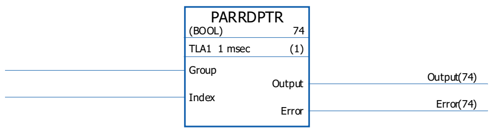
<table>
<tr><th colspan="2">GetBitPtr<br>(10099)</th></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>-</td></tr>
<tr><td>Operation</td><td>Reads the status of one bit within a parameter value cyclically.<br>The Bit ptr input specifies the parameter group, index and bit to be read.<br>The output (Out) provides the value of the bit.</td></tr>
<tr><td>Inputs</td><td>Parameter group, index and bit (Bit ptr): DINT</td></tr>
<tr><td>Outputs</td><td>Bit status (Out): DINT</td></tr>
</table>

<table>
<tr><th colspan="2">GetValPtr<br>(10098)</th></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>-</td></tr>
<tr><td>Operation</td><td>Reads the value of a parameter cyclically.<br>The Par ptr input specifies the parameter group and index to be read.<br>The output (Out) provides the value of the parameter.</td></tr>
<tr><td>Inputs</td><td>Parameter group and index (Par ptr): DINT</td></tr>
<tr><td>Outputs</td><td>Parameter value (Out): DINT</td></tr>
</table>

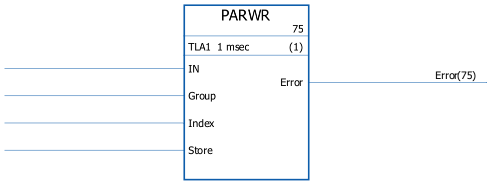
<table>
<tr><th colspan="2">PARRD<br>(10082)</th></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>6.00 µs</td></tr>
<tr><td>Operation</td><td>Reads the scaled value of a parameter (specified by the Group and Index inputs). If the parameter is a pointer parameter, the Output pin provides the number of the source parameter instead of its value.<br>Error codes are indicated by the error output (Error) as follows:
<table>
<tr><th>Error code</th><th>Description</th></tr>
<tr><td>0</td><td>No error</td></tr>
<tr><td><> 0</td><td>Error</td></tr>
</table>
See also blocks PARRDINTR and PARRDPTR.</td></tr>
</table>

| | |
|---|---|
| **Inputs** | Parameter group input (Group): DINT<br>Parameter index input (Index): DINT |
| **Outputs** | Output (Output): DINT<br>Error output (Error): DINT |

<table>
<tr><th colspan="2">PARRDINTR<br>(10101)</th></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>-</td></tr>
<tr><td>Operation</td><td>Reads the internal (non-scaled) value of a parameter (specified by the Group and Index inputs). The value is provided by the Output pin.<br>Error codes are indicated by the error output (Error) as follows:
<table>
<tr><th>Error code</th><th>Description</th></tr>
<tr><td>0</td><td>No error or busy</td></tr>
<tr><td><> 0</td><td>Error</td></tr>
</table>
Note: Using this block may cause incompatibility issues when upgrading the application to another firmware version.</td></tr>
<tr><td>Inputs</td><td>Parameter group (Group): DINT<br>Parameter index (Index): DINT</td></tr>
<tr><td>Outputs</td><td>Output (Output): Boolean, INT, DINT, REAL, REAL24<br>Error output (Error): DINT</td></tr>
</table>

<table>
<tr><th colspan="2">PARRDPTR<br>(10100)</th></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>-</td></tr>
<tr><td>Operation</td><td>Reads the internal (non-scaled) value of the source of a pointer parameter. The pointer parameter is specified using the Group and Index inputs.<br>The value of the source selected by the pointer parameter is provided by the Output pin.<br>Error codes are indicated by the error output (Error) as follows:
<table>
<tr><th>Error code</th><th>Description</th></tr>
<tr><td>0</td><td>No error or busy</td></tr>
<tr><td><> 0</td><td>Error</td></tr>
</table></td></tr>
</table>

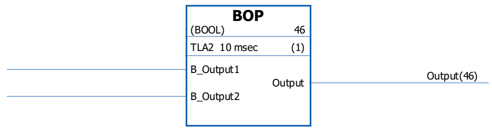
| | |
|---|---|
| **Inputs** | Parameter group (Group): DINT<br>Parameter index (Index): DINT |
| **Outputs** | Output (Output): Boolean, INT, DINT, REAL, REAL24<br>Error output (Error): DINT |

<table>
<tr><td colspan="2">PARWR<br>(10080)</td></tr>
<tr><td>Illustration</td><td>
</td></tr>
<tr><td>Execution time</td><td>14.50 µs</td></tr>
<tr><td>Operation</td><td>The input value (IN) is written to the defined parameter (Group and Index).<br>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.<br>Error codes are indicated by the error output (Error) as follows:
<table>
<tr><th>Error code</th><th>Description</th></tr>
<tr><td>0</td><td>No error</td></tr>
<tr><td><> 0</td><td>Error</td></tr>
</table></td></tr>
<tr><td>Inputs</td><td>Input (IN): DINT<br>Parameter group input (Group): DINT<br>Parameter index input (Index): DINT<br>Store input (Store): Boolean</td></tr>
<tr><td>Outputs</td><td>Error output (Error): DINT</td></tr>
</table>

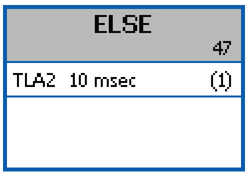
# Program structure

## BOP

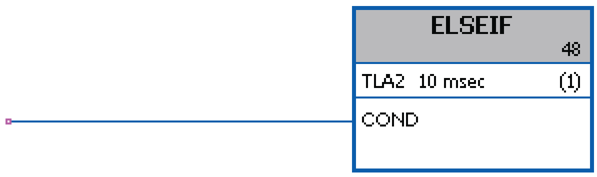
(10105)

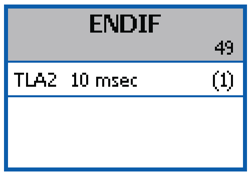
| | |
|---|---|
| **Illustration** |  |
| **Execution time** | - |
| **Operation** | The BOP (Bundle OutPut) block collects the outputs of several different sources. The sources are connected to the B_Output pins. The B_Output pin that changed last is relayed to the Output pin.<br>The block is intended for use with conditional IF-ENDIF structures. See the example under the *IF* block. |
| **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 |

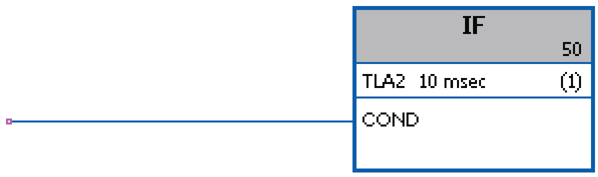
## ELSE

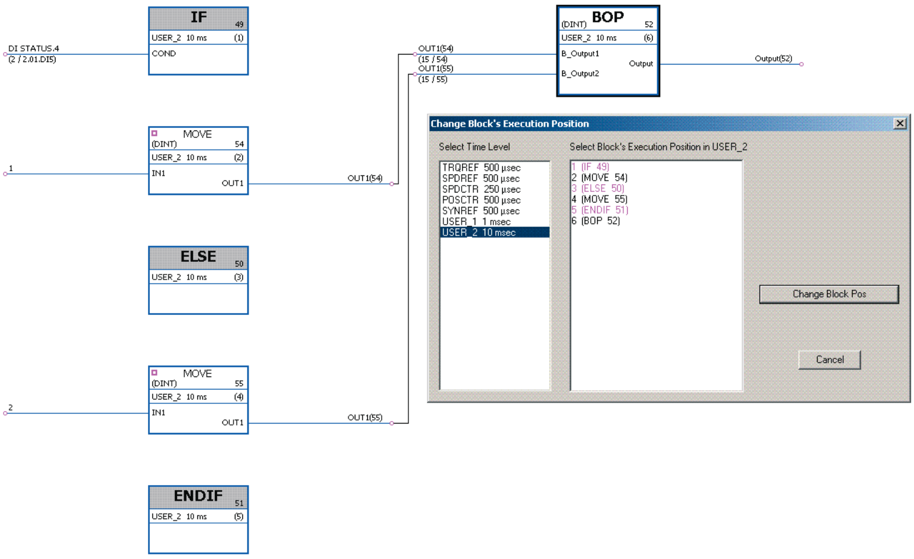
| | |
|---|---|
| **Illustration** |  |
| **Execution time** | - |
| **Operation** | See the description of the *IF* block. |
| **Inputs** | - |
| **Outputs** | - |

## ELSEIF

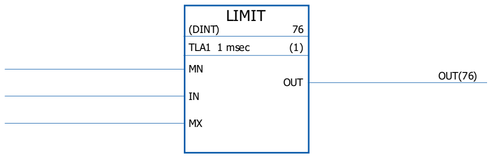
| | |
|---|---|
| **Illustration** |  |
| **Execution time** | - |
| **Operation** | See the description of the *IF* block. |
| **Inputs** | Input (COND): Boolean |
| **Outputs** | - |

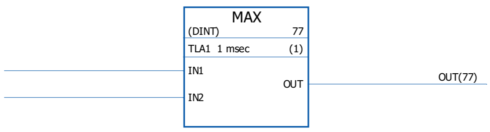
| **ENDIF** | |
|---|---|
| **Illustration** |  |
| **Execution time** | - |
| **Operation** | See the description of the *IF* block. |
| **Inputs** | - |
| **Outputs** | - |

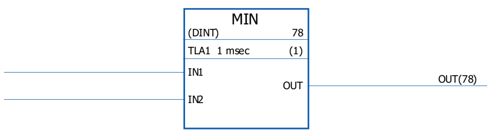
| **IF**<br><br>(10103) | |
|---|---|
| **Illustration** |  |
| **Execution time** | - |

| | |
|---|---|
| **Operation** | The IF, ELSE, ELSEIF and ENDIF blocks define, by Boolean logic, which parts of the application program are executed.<br>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.<br>The outputs of the “branches” are collected and selected by using the *BOP* block.<br>Example:<br>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.<br> |
| **Inputs** | Input (COND): Boolean |
| **Outputs** | - |

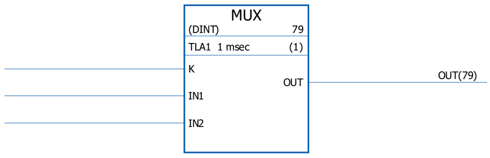
# Selection

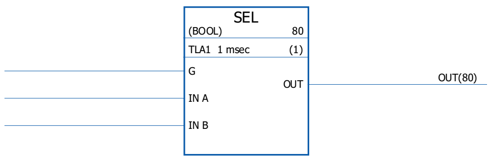
| **LIMIT**<br>(10052) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.53 µs |
| **Operation** | The output (OUT) is the limited input (IN) value. Input is limited according to the minimum (MN) and maximum (MX) values. |
| **Inputs** | The input data type is selected by the user.<br>Minimum input limit (MN): INT, DINT, REAL, REAL24<br>Input (IN): INT, DINT, REAL, REAL24<br>Maximum input limit (MX): INT, DINT, REAL, REAL24 |
| **Outputs** | Output (OUT): INT, DINT, REAL, REAL24 |

| **MAX**<br>(10053) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.81 µs (when two inputs are used) + 0.53 µs (for every additional input). When all inputs are used, the execution time is 16.73 µs. |
| **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.<br>Input (IN1…IN32): INT, DINT, REAL, REAL24 |
| **Outputs** | Output (OUT): INT, DINT, REAL, REAL24 |

| **MIN**<br>(10054) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.81 µs (when two inputs are used) + 0.52 µs (for every additional input). When all inputs are used, the execution time is 16.50 µs. |
| **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.<br>Input (IN1…IN32): INT, DINT, REAL, REAL24 |

| | |
|---|---|
| **Outputs** | Output (OUT): INT, DINT, REAL, REAL24 |

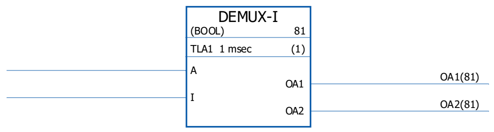
| **MUX**<br>(10055) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.70 µs |
| **Operation** | The value of an input (IN) selected by the address input (K) is stored to the output (OUT).<br>If the address input is 0, negative or exceeds the number of the inputs, the output is 0. |
| **Inputs** | The input data type and number of inputs (2…32) are selected by the user.<br>Address input (K): DINT<br>Input (IN1…IN32): INT, DINT, REAL, REAL24 |
| **Outputs** | Output (OUT): INT, DINT, REAL, REAL24 |

| **SEL**<br>(10056) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 1.53 µs |
| **Operation** | The output (OUT) is the value of the input (IN) selected by the selection input (G).<br>If G = 0: OUT = IN A.<br>If G = 1: OUT = IN B. |
| **Inputs** | The input data type is selected by the user.<br>Selection input (G): Boolean<br>Input (IN A, IN B): Boolean, INT, DINT, REAL, REAL24 |
| **Outputs** | Output (OUT): Boolean, INT, DINT, REAL, REAL24 |

# Switch & Demux

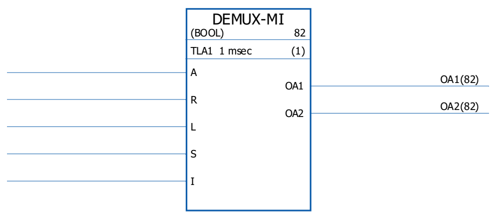
## DEMUX-I

(10061)

| | |
|---|---|
| **Illustration** |   |
| **Execution time** | 1.38 µs (when two outputs are used) + 0.30 µs (for every additional output). When all outputs are used, the execution time is 10.38 µs. |
| **Operation** | Input (I) value is stored to the output (OA1…OA32) selected by the address input (A). All other outputs are 0.<br>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.<br>Address input (A): DINT<br>Input (I): INT, DINT, Boolean, REAL, REAL24 |
| **Outputs** | The number of the output channels (1…32) is selected by the user.<br>Output (OA1…OA32): INT, DINT, REAL, REAL24, Boolean |

## DEMUX-MI

(10062)

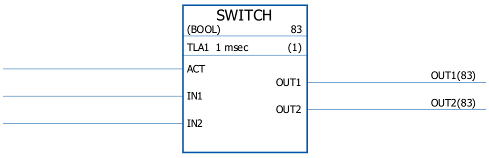
| | |
|---|---|
| **Illustration** |   |
| **Execution time** | 0.99 µs (when two outputs are used) + 0.25 µs (for every additional output). When all outputs are used, the execution time is 8.4 µs. |
| **Operation** | The input (I) value is stored to the output (OA1…OA32) selected by the address input (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.<br>If the reset input (R) is 1, all connected outputs are 0.<br>If the address input is 0, negative or exceeds the number of the outputs, all outputs are 0.<br>Example: |

| 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.<br>Address input (A): DINT<br>Reset input (R): Boolean<br>Load input (L): Boolean<br>Set input (S): Boolean<br>Input (I): DINT, INT, REAL, REAL24, Boolean |
| **Outputs** | The number of the output channels (1…32) is selected by the user.<br>Output (OA1…OA32): DINT, INT, REAL, REAL24, Boolean |

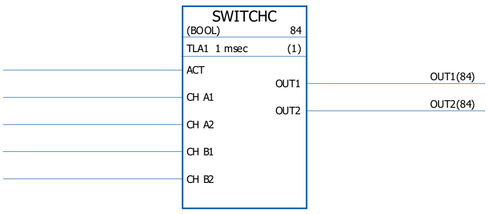
## SWITCH

(10063)

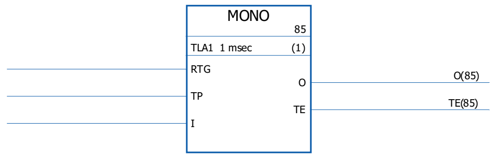
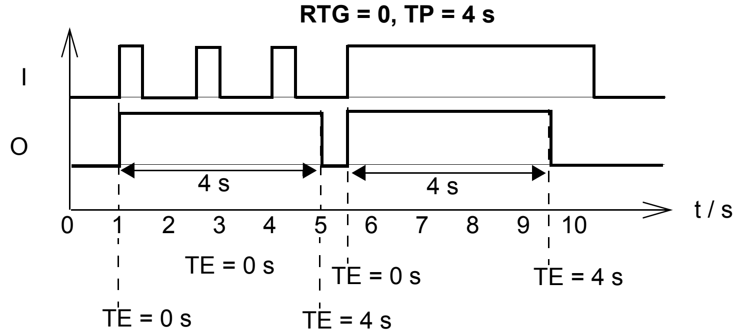
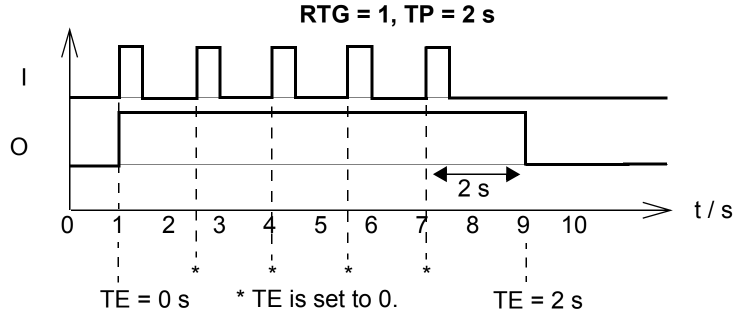
| | |
|---|---|
| **Illustration** |  |
| **Execution time** | 0.68 µs (when two inputs are used) + 0.50 µs (for every additional input). When all inputs are used, the execution time is 15.80 µs. |
| **Operation** | The output (OUT) is equal to the corresponding input (IN) if the activate input (ACT) is 1. Otherwise the output is 0. |
| **Inputs** | The input data type and the number of inputs (1…32) are selected by the user.<br>Activate input (ACT): Boolean<br>Input (IN1…IN32): INT, DINT, REAL, REAL24, Boolean |
| **Outputs** | Output (OUT1…OUT32): INT, DINT, REAL, REAL24, Boolean |

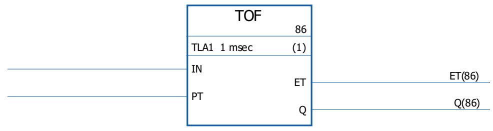
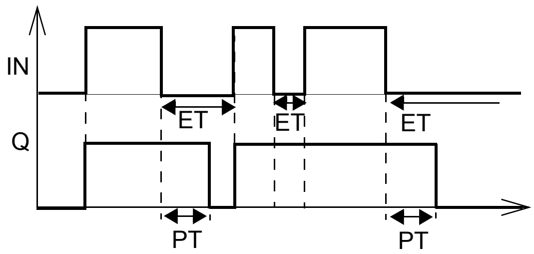
## SWITCHC

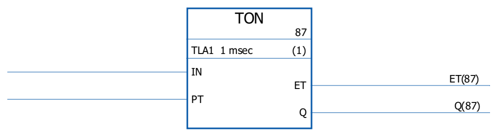
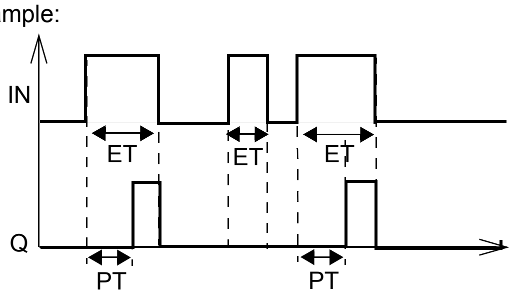
(10064)

| | |
|---|---|
| **Illustration** |  |
| **Execution time** | 1.53 µs (when two inputs are used) + 0.73 µs (for every additional input). When all inputs are used, the execution time is 23.31 µs. |
| **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.<br>Activate input (ACT): Boolean<br>Input (CH A1…CH A32, CH B1…CH B32): INT, DINT, REAL, REAL24, Boolean |
| **Outputs** | Output (OUT1…OUT32): INT, DINT, REAL, REAL24, Boolean |

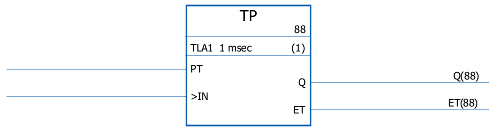
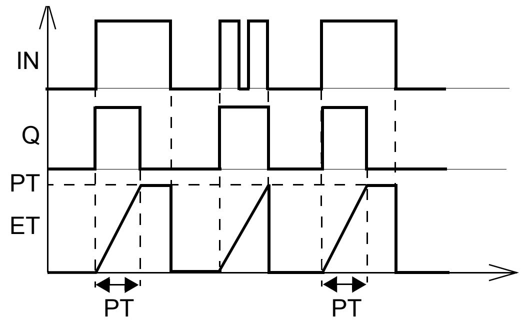
# Timers

| **MONO**<br>(10057) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 1.46 µs |
| **Operation** | 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.<br>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.<br>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.<br>Example 1: MONO is not re-triggable, i.e. RTG = 0.<br>**RTG = 0, TP = 4 s**<br><br>Example 2: MONO is re-triggable, i.e. RTG = 1.<br>**RTG = 1, TP = 2 s**<br> |
| **Inputs** | Re-trigger input (RTG): Boolean<br>Time pulse input (TP): DINT (1 =  µs)<br>Input (I): Boolean |
| **Outputs** | Output (O): Boolean<br>Time elapsed output (TE): DINT (1 = 1 µs) |

| **TOF**<br>(10058) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 1.10 µs |
| **Operation** | 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).<br>Elapsed time count (ET) starts when the input is set to 0 and stops when the input is set to 1.<br>Example:<br> |
| **Inputs** | Input (IN): Boolean<br>Pulse time input (PT): DINT (1 = 1 µs) |
| **Outputs** | Elapsed time output (ET): DINT (1 = 1 µs)<br>Output (Q): Boolean |

| **TON**<br>(10059) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 1.22 µs |
| **Operation** | 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.<br>Elapsed time count (ET) starts when the input is set to 1 and stops when the input is set to 0.<br>Example:<br> |

| | |
|---|---|
| **Inputs** | Input (IN): Boolean<br>Pulse time input (PT): DINT (1 = 1 µs) |
| **Outputs** | Elapsed time output (ET): DINT (1 = 1 µs)<br>Output (Q): Boolean |

| **TP**<br>(10060) | |
|---|---|
| **Illustration** |  |
| **Execution time** | 1.46 µs |
| **Operation** | 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).<br>Elapsed time count (ET) starts when the input is set to 1 and stops when the input is set to 0.<br> |
| **Inputs** | Pulse time input (PT): DINT (1 = 1 µs)<br>Input (IN): Boolean |
| **Outputs** | Output (Q): Boolean<br>Elapsed time output (ET): DINT (1 = 1 µs) |

5

# 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.

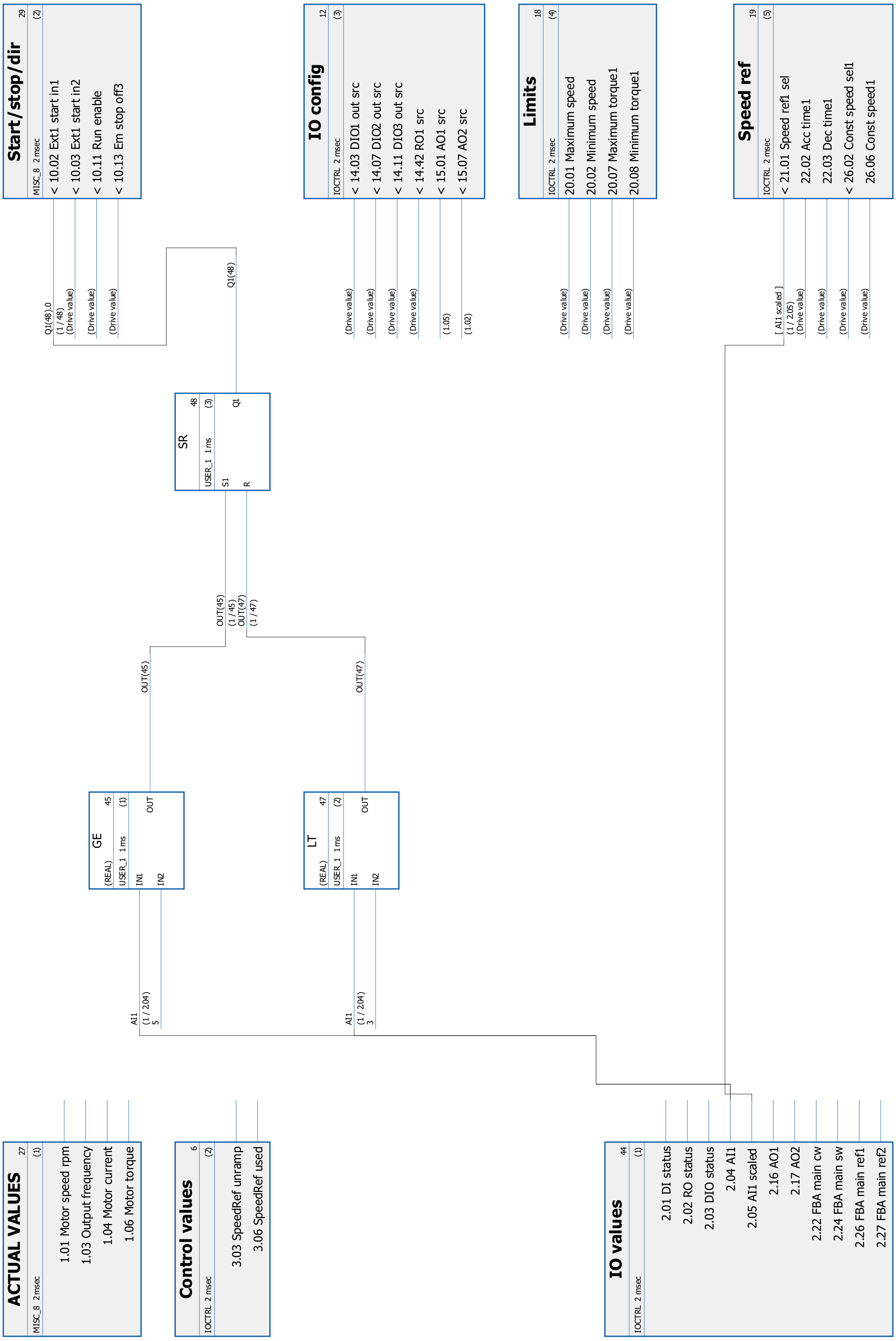
## Start/stop using analog input

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.


Start/stop/dir
MISC_8 2 msec
< 10.02 Ext1 start in1
< 10.03 Ext1 start in2
< 10.11 Run enable
< 10.13 Em stop off3
IO config
IOCTRL 2 msec
< 14.03 DIO1 out src
< 14.07 DIO2 out src
< 14.11 DIO3 out src
< 14.42 RO1 src
< 15.01 AO1 src
< 15.07 AO2 src
Limits
IOCTRL 2 msec
20.01 Maximum speed
20.02 Minimum speed
20.07 Maximum torque1
20.08 Minimum torque1
Speed ref
IOCTRL 2 msec
< 21.01 Speed ref1 sel
22.02 Acc time1
22.03 Dec time1
< 26.02 Const speed sel1
26.06 Const speed1
SR
USER_1 1 ms
S1
R
Q1
Q1(48)
Q1(48).0
OUT(45)
OUT(47)
GE
(REAL)
USER_1 1 ms
IN1
IN2
OUT
LT
(REAL)
USER_1 1 ms
IN1
IN2
OUT
AI1
ACTUAL VALUES
MISC_8 2 msec
1.01 Motor speed rpm
1.03 Output frequency
1.04 Motor current
1.06 Motor torque
Control values
IOCTRL 2 msec
3.03 SpeedRef unramp
3.06 SpeedRef used
IO values
IOCTRL 2 msec
2.01 DI status
2.02 RO status
2.03 DIO status
2.04 AI1
2.05 AI1 scaled
2.16 AO1
2.17 AO2
2.22 FBA main cw
2.24 FBA main sw
2.26 FBA main ref1
2.27 FBA main ref2

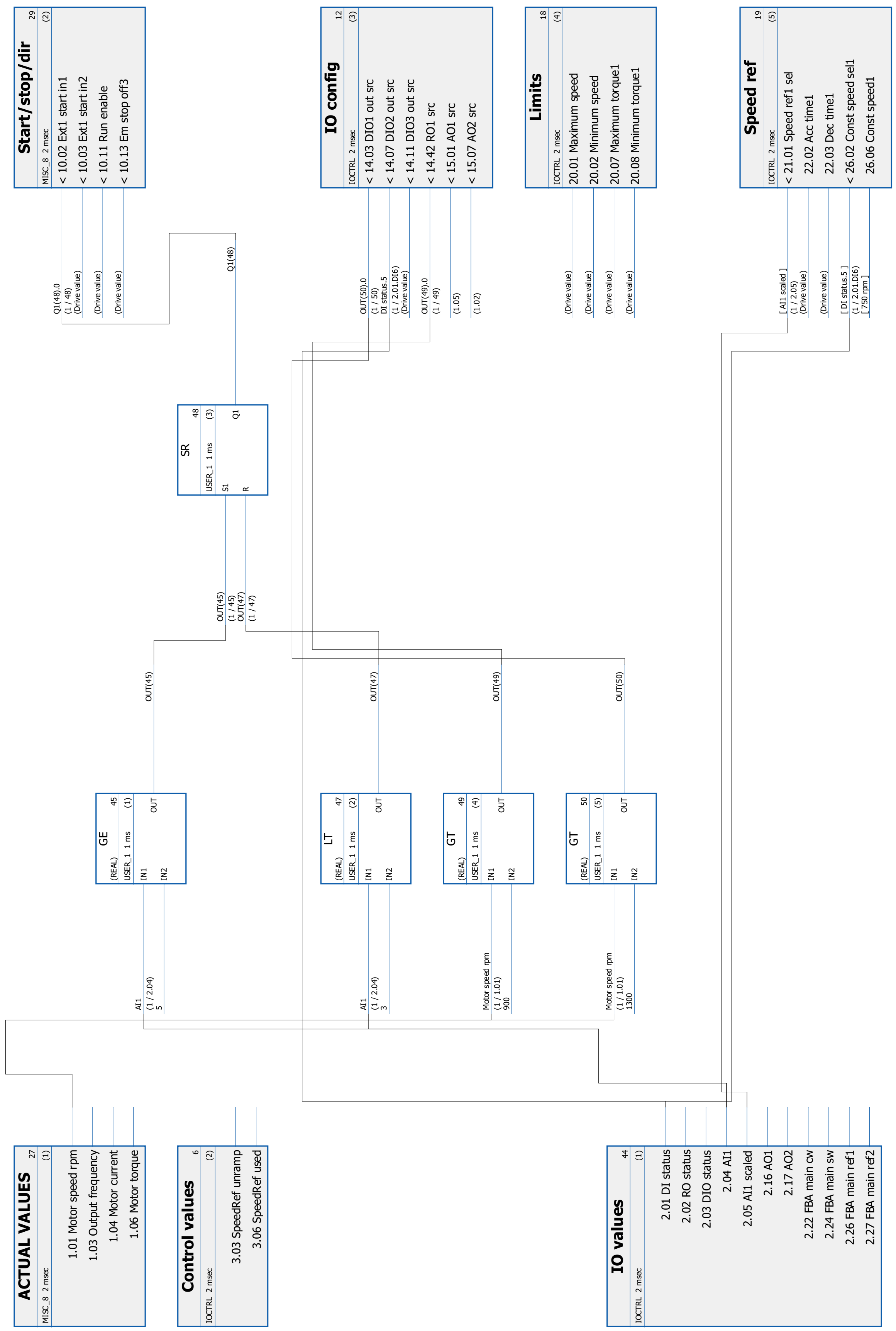
## Relay output and digital input/output control

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.

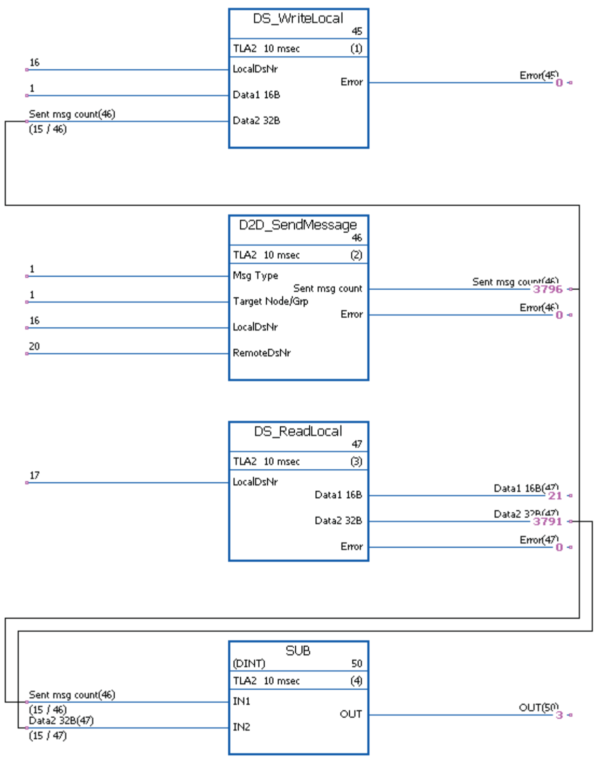

Start/stop/dir
MISC_8 2 msec 29 (2)
< 10.02 Ext1 start in1
< 10.03 Ext1 start in2
< 10.11 Run enable
< 10.13 Em stop off3
IO config
IOCTRL 2 msec 12 (3)
< 14.03 DIO1 out src
< 14.07 DIO2 out src
< 14.11 DIO3 out src
< 14.42 RO1 src
< 15.01 AO1 src
< 15.07 AO2 src
Limits
IOCTRL 2 msec 18 (4)
20.01 Maximum speed
20.02 Minimum speed
20.07 Maximum torque1
20.08 Minimum torque1
Speed ref
IOCTRL 2 msec 19 (5)
< 21.01 Speed ref1 sel
22.02 Acc time1
22.03 Dec time1
< 26.02 Const speed sel1
26.06 Const speed1
SR
USER_1 1 ms 48 (3)
S1
R
Q1
GE
LT
GT
GT
(REAL)
USER_1 1 ms
IN1
IN2
OUT
ACTUAL VALUES
MISC_8 2 msec 27 (1)
1.01 Motor speed rpm
1.03 Output frequency
1.04 Motor current
1.06 Motor torque
Control values
IOCTRL 2 msec 6 (2)
3.03 SpeedRef unramp
3.06 SpeedRef used
IO values
IOCTRL 2 msec 44 (1)
2.01 DI status
2.02 RO status
2.03 DIO status
2.04 AI1
2.05 AI1 scaled
2.16 AO1
2.17 AO2
2.22 FBA main cw
2.24 FBA main sw
2.26 FBA main ref1
2.27 FBA main ref2

# Drive-to-drive communication (ACS850 only)

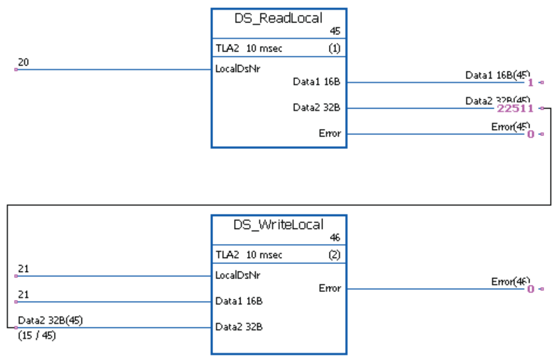
For the descriptions of the drive-to-drive standard function blocks, see section *Communication (ACS850 only)* on page 31.

## Example of master point-to-point messaging

*Master*



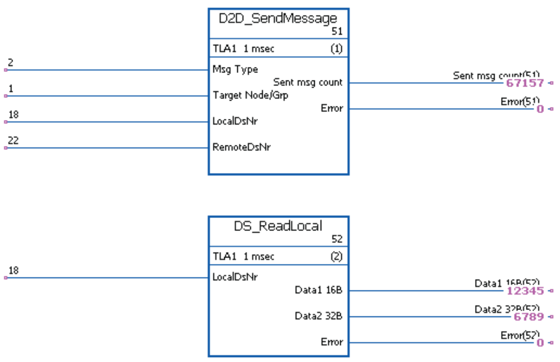

*Follower (node 1)*



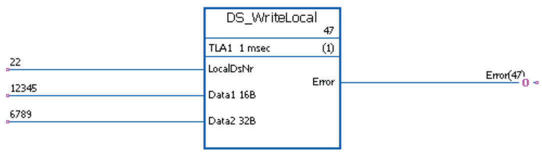

1. The master sends a constant (1) and the value of the message counter into follower dataset 20. Data is prepared to and sent from dataset 16.
2. The follower sends the received counter value and a constant (21) as a reply to the master.
3. The master calculates the difference of the latest message number and received data.

## Example of read remote messaging

*Master*



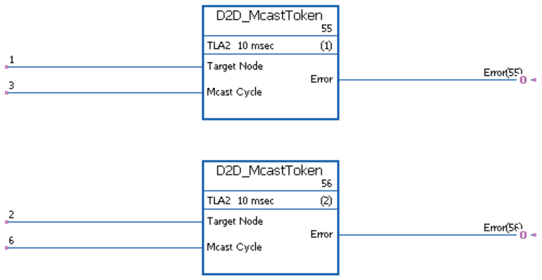

*Follower (node 1)*




1. The master reads the contents of the follower dataset 22 into its own dataset 18. Data is accessed using the *DS_Read_Local* block.
2. In the follower, constant data is prepared into dataset 22.

## Example of releasing tokens for follower-to-follower communication

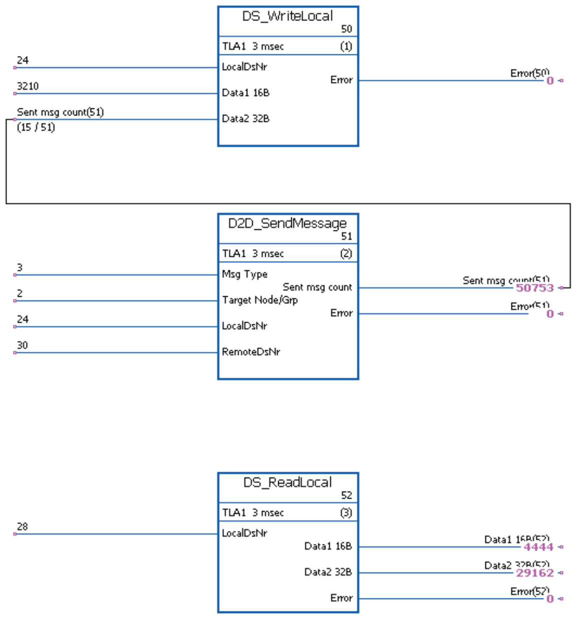
*Master*



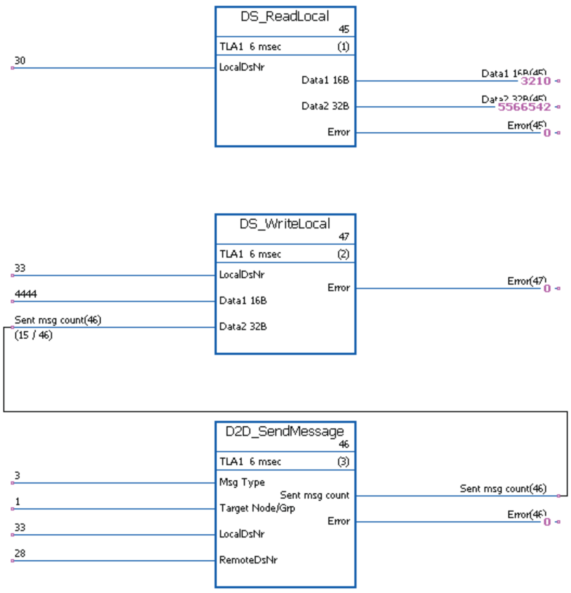

1. This drive-to-drive link consists of three drives (master and two followers).
2. The master operates as a “chairman”. Follower 1 (node 1) is allowed to send one message every 3 milliseconds. Follower 2 (node 2) is allowed to send one message every 6 milliseconds.

## Example of follower point-to-point messaging

*Follower 1 (node 1)*



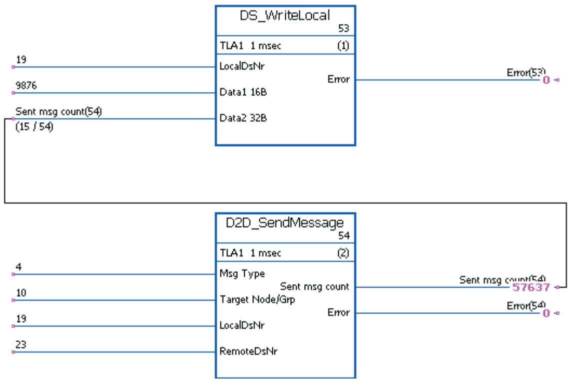

*Follower 2 (node 2)*



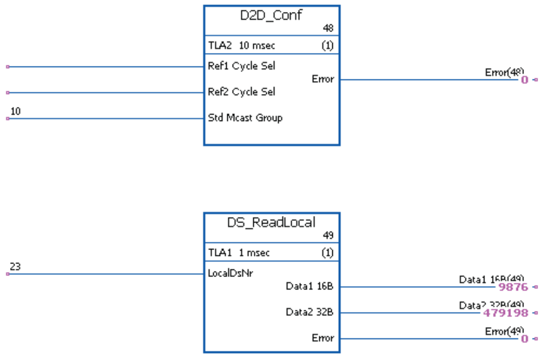

1. Follower 1 writes local dataset 24 to follower 2 dataset 30 (3 ms interval).
2. Follower 2 writes local dataset 33 to follower 1 dataset 28 (6 ms interval).
3. In addition, both followers read received data from local datasets.

## Example of standard multicast messaging

*Master*




*Follower(s) in Std Mcast Group 10*

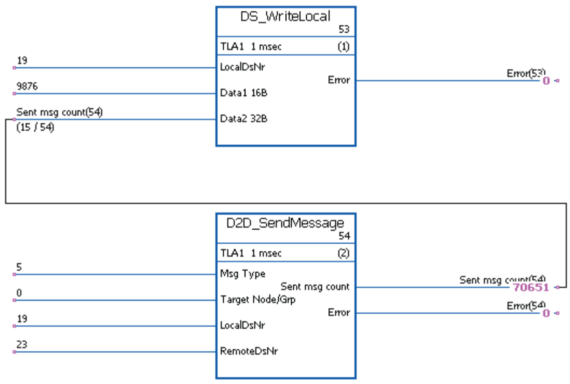



1. The master sends a constant (9876) and the value of the message counter to all followers in standard multicast group 10. The data is prepared into and sent from master dataset 19 to follower dataset 23.
2. Received data is read from dataset 23 of the receiving followers.

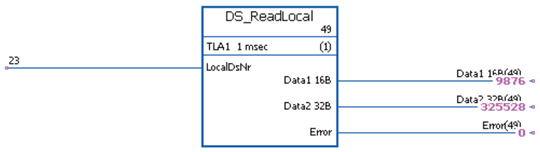
**Note:** The example application shown for Master above also applies to the sending follower in standard follower-to-follower multicasting.

## 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

3AUA0000078664 Rev B EN 2011-11-28