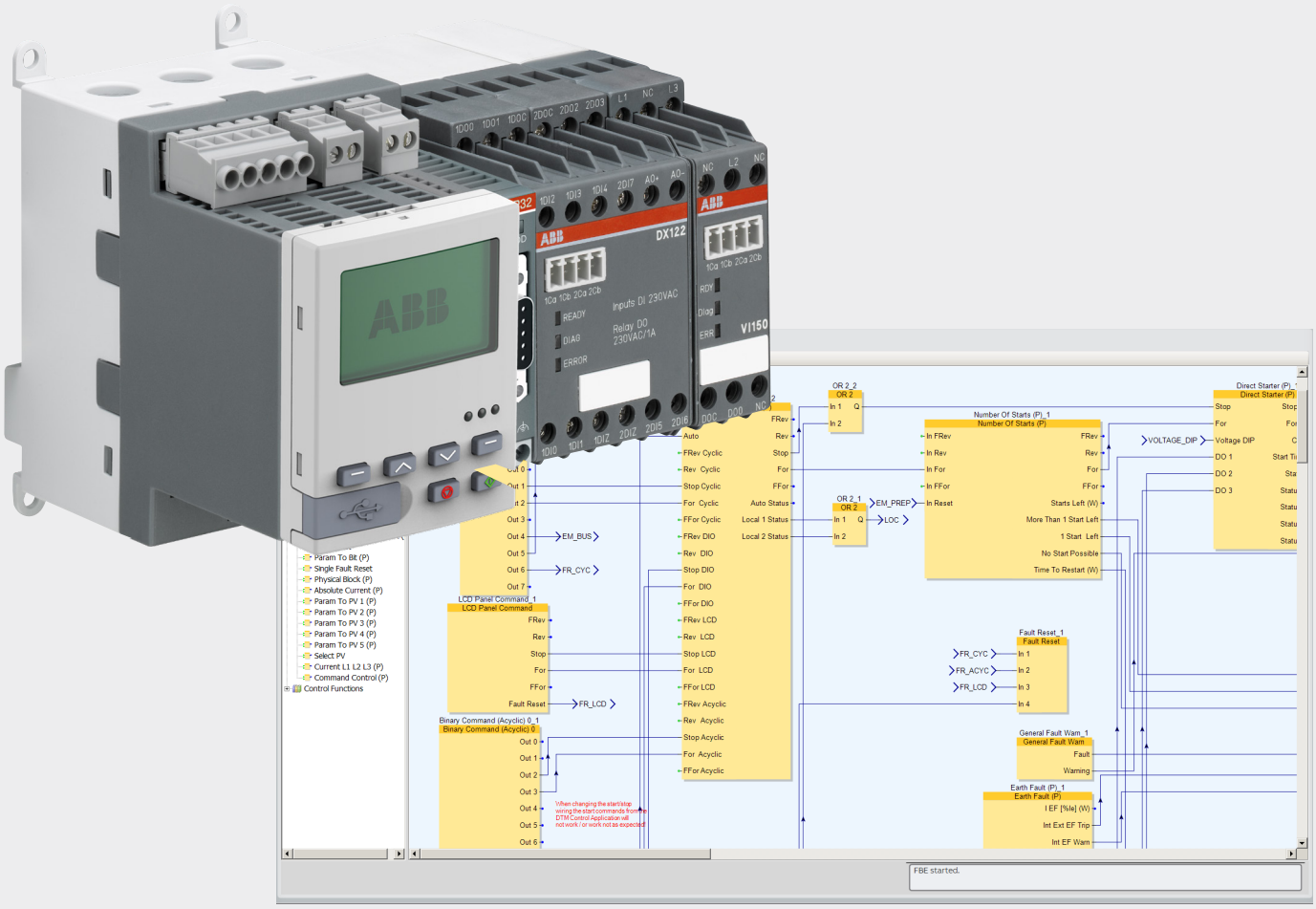
MANUAL

Custom Application Editor

Universal Motor Controller UMC100.3



Important notice

Target group
This description is intended for the use of trained specialists in electrical installation and control and automation engineering, who are familiar with the applicable national standards.

Safety requirements
The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Using this handbook

Symbols
This technical document contains sentinels to point the reader to important information, potential risks and precaution information. The following symbols are used:

	Symbol to indicate a potentially dangerous situation that can cause damage to the UMC or connected devices or to the environment.
	Symbol to indicate important information and conditions.
	Symbol that indicates a potentially dangerous situation that can cause personal injury

Terms and abbreviations

FDI	Field Device Integration Technology. This is a system-wide solution. Host systems, like asset management systems, configurators, and device managers must support the FDI client-server architecture. Field device suppliers must encapsulate device specific software and documentation into an FDI Device Package. See https://www.fieldcommgroup.org/ for more information
FIM	Field Information Manager. ABB software for the configuration of field devices mainly in process industry. FIM s based on the FDI standard
FIM UMC Edition	Dedicated FIM version for the configuration of UMC100.3 There are also other versions of FIM available, that also enable the UMC100.3 configuration. Following we refer always to this version. Sometimes this version is only called FIM
FIM Device Window Edition	Limited trial version version of the software, free of charge. It also includes the Custom Application Editor, but no offline mode for saving the settings on the PC
GSD	Geräte Stammdatei (German) which means a electronic data sheet of a device
DCS	Distributed Control System
PLC	Programmable Logic Controller

Related documents

Name	Type	Document No.
UMC100.3 Universal Motor Controller	Manual English version	2CDC135032D0204
UMC100.3 Universal Motor Controller ATEX	Manual English version	2CDC135033D0202

Table of contents

Table of contents

Table of contents

How to get started

There are many options for using the UMC100.3. Not all functions are required in all cases. Therefore the documentation is split into separate parts. You only need to read the parts that are relevant for your application.
The UMC100.3 documentation is split into the following manuals.

UMC100.3 Technical Manual (2CDC135032D0204)
This is the main technical description and must be read carefully

PBDTM (2CDC192012D0209)
This manual describes the configuration tool
(Device Type Manager) that can be used to configure and monitor the UMC100.3 (V6.0).
Use version V5.0 for other ABB control products like UMC100.3, UMC22, softstarters and circuit breakers .
It is based on the standardised FDT/DTM technology.
If you plan to only configure the UMC100 with the device description files like GSD (for PROFIBUS) or EDS (for DeviceNet) or via the LCD panel you do not need to read this manual.

UMC100.3 Custom Application Editor (2CDC135034D0201)
This manual describes how to create customer-specific applications for the UMC100. You only need to read this manual if the logic built into the UMC100 does not fulfill your needs. As the custom application editor is part of the configuration tool (Device Type Manager) it is highly recommended that you read the PBDTM manual first.

You are now reading this manual.

FIM UMC Edition (2CDC135075M0201)
This manual describes the configuration software FIM UMC Edition that can be used to configure and monitor the UMC100.3.
This software is based on the FDI (Field Device Integration) standard for the configuration of field devices.
FIM UMC Edition replaces the former PBDTM software.

If you plan to only configure the UMC100.3 with device description files like GSD (PROFIBUS), GSDML (Profinet) or EDS (DeviceNet, EtherNet/IP™) or via the UMC100-PAN control panel you do not need to read this manual.

Fieldbus and Ethernet Communication Interfaces
Whenever the UMC100.3 is to be connected to a fieldbus or Ethernet network, read the appropriate manual. Presently communication interfaces for PROFIBUS DP (PDP32.0), Modbus RTU (MRP31.0) and DeviceNet™ (DNP31.0) exist. For Ethernet there are EtherNet/IP™ (EIU32.0), Modbus TCP (MTQ22-FBP.0) and Profinet (PNQ22-FBP.0) available.
The UMC100.3 can be used together with these communication interfaces.

What is new in this version

Library versions

Each UMC version got additional expansion modules and new function blocks. The library version setting for UMC100.3 is version 3.0, which includes all function blocks from former versions. All versions are upwards compatible to newer UMC100 releases.

The version number is stored together with the logic.

V1.0	UMC100-FBP.0	1SAJ520000R0100, 1SAJ520000R0200
V2.0	UMC100-FBP.0	1SAJ520000R0101, 1SAJ520000R0201
V3.0	UMC100.3	all versions

For UMC100.3 generally V3.0 should be selected as shown

Backward compatibility

The new UMC100.3 can execute applications created with the previous library version V1.0 and V2.0 without any changes.

With the latest custom application editor you can still create applications for previous UMC versions.

If you import existing applications created with the function block library version 1.0 or 2.0 the library selector is set to 1.0 or 2.0 automatically. If you want to use new function blocks in your application change the library version to 3.0 now and add the new function blocks. This application can not be used in an UMC R0100xR020x anymore but only in the new UMC100.3.

Consideration on parameter I/O data profile

With the parameter "I/O data profile" it is possible to change the length of the cyclically transmitted I/O data.

In case this parameter is set to "Profile 2", the data of the following function blocks are not sent to or received from the bus:

- Analog command 0
- Binary command 2/3
- Binary monitoring 2/3/4/5
- Analog monitoring 1/2/3/4

It is not necessary to delete these blocks from a diagram.

Introduction

The UMC100.3 offers several built-in applications which each realise another control function. Built-in applications cannot be changed. But their behavior can be widely adapted to your needs simply by changing parameters. Those parameters are described in the UMC100.3 manual.

One application is reserved is reserved for a custom-specific logic and is empty by default. This application has to be created with the Custom Application Editor and then be downloaded to the UMC100.3 It is then stored in the UMC's non-volatile memory.

The description of the Custom Application Editor and the function blocks is the subject of this manual.
The editor itself is part of the configuration software FIM UMC Edition. Installation and handling of this software are described in a separate manual.

—

Recommendations

Before creating any custom application, carefully read the UMC100.3 manual and check whether the requirements can be fulfilled using the standard built-in applications. Standard applications are simpler to use because no programming is involved.

In case any custom application is required, all control functions are available as templates. They can be imported in the editor and modified as required. This has the advantage, that the templates are tested and working. Tests are only required for the modifications. Function blocks that are not required (e.g. for expansion modules) do not need to be removed.

Of course it is also possible to create an individual logic starting from zero. All required function blocks are available in the library. It can be helpful to have before a look to the templates and trying to understand how the predefined control functions are working.

Before you create your own custom application carefully read the UMC100 manual and check whether your requirements can be fulfilled using the standard built-in applications.
Standard applications are simpler to use because no programming is involved.

—

The Editor

The Custom Application Editor offers the possibility to create individual control applications in a graphically based programming language. The editor permits simple positioning and connecting of function blocks. The function blocks are oriented on the IEC 61131-3 standard where applicable.

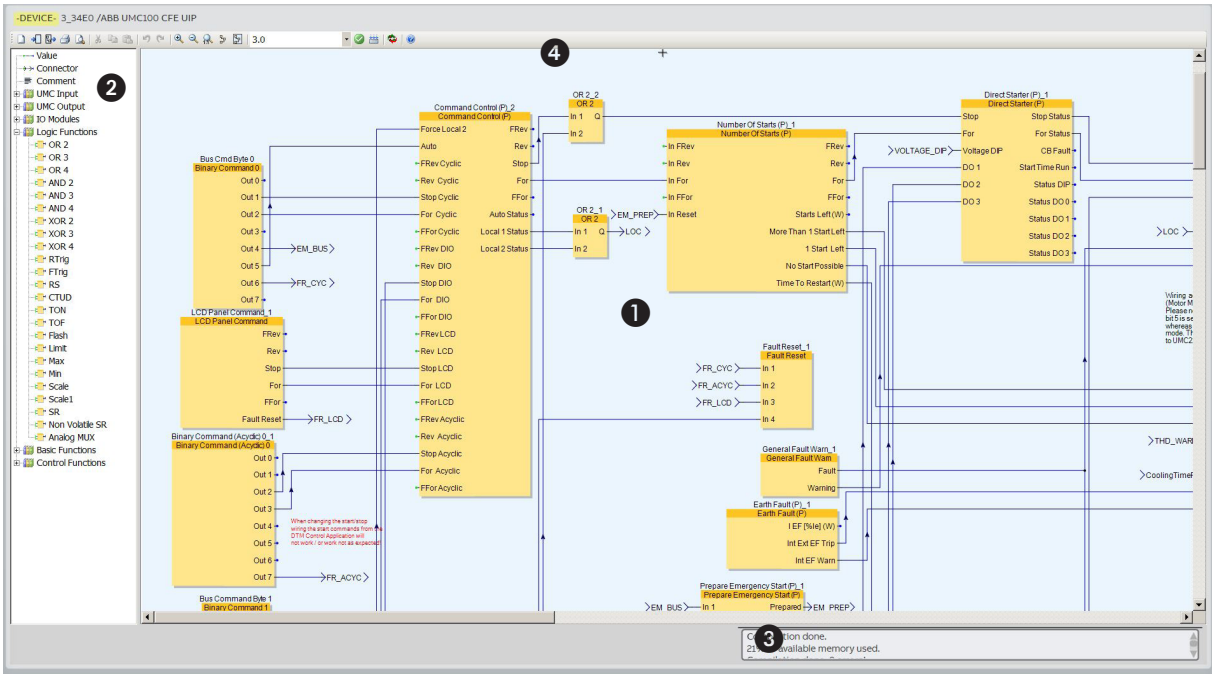
Figure 1 shows the FIM software with the Custom Application Editor. The editor is part of the Device Settings / Parametrize menu, where also the UMC100.3 parameters are set.

The function blocks and signal flow lines are programmed in the worksheet ❶. You can place the elements from the library window ❷ everywhere on the worksheet. Parameter values are edited in rectangles. On the top of the editor window the toolbar (4) allows quick access to the most important commands.

If you have created an application a checker performs several consistency checks to ensure that the application is valid and can be executed by the UMC. If there are any problems detected during the check they are shown in the Message Window (3).

One worksheet contains the whole customer-defined application. The individual areas can be accessed via vertical and horizontal scrolling. The printed form of the program contains page-for-page exactly what is seen on the screen.

Introduction



The window of the configuration tool with active Custom Application Editor consists of:
❶ Worksheet, ❷ Function Block Library, ❸ Message Window, ❹ Custom Editor Toolbar,
❺ selector for the function block library version.

An application program consists of the following graphical elements:
- Connections (lines)
- Parameter values
- Function blocks
- Comments
- Connectors

The signal flow of an application is from left to right. The signal flow lines are edited with the left mouse button. Elements can be dragged from the library window ❷ and dropped in the worksheet ❶ .
During commissioning (when an online connection to the UMC exists) the present application status can be displayed.

How to proceed
It is highly recommended that you read chapter "Application Elements" to gain a basic understanding of the elements an application consists of. Then the section "First Application" should be consulted when starting up.

Application elements

This section presents the different elements a custom application is made of.

Function Blocks

Function blocks enclose a function. When being executed a function block takes the input values and calculates the output values. Function blocks in the UMC can contain status information (e.g. the starter function blocks). This means that whenever a function is retrieved with the same arguments (input parameters) the same result (output parameter) is not necessarily obtained.

The block frame limits the selector area of the block. From a coloured green line around the type name you can establish whether the block has been selected or not (see diagram below). If there is no green line the block is not selected. If you move over a block with the mouse and stay there for a moment a tool tip is displayed presenting a short description of the block.

Function block name and instance name:
All function blocks are displayed with the block type name which describes its type. This text cannot be changed. Above the block the instance name is shown which is automatically determined when placing a new block. This name can be changed but must be unique within the application. In the example below the type of the left block is T_{ON} (= On Delay) whereas the user-defined instance name was changed to "Start Delay" which describes the function of this block in the application.

Input/output pins:
A distinction must be made here between inputs and outputs. In accordance with the signal flow, inputs are always displayed on the left and outputs on the right. IO pins are also known as terminals.

Terminal designation:
In a function block each input/output terminal also has a code that represents the function of the terminal, e.g. EN for enable, Q for an output etc. If you move over a terminal and stay there for a moment a tool tip text is displayed presenting the data type and a short description of the terminal.

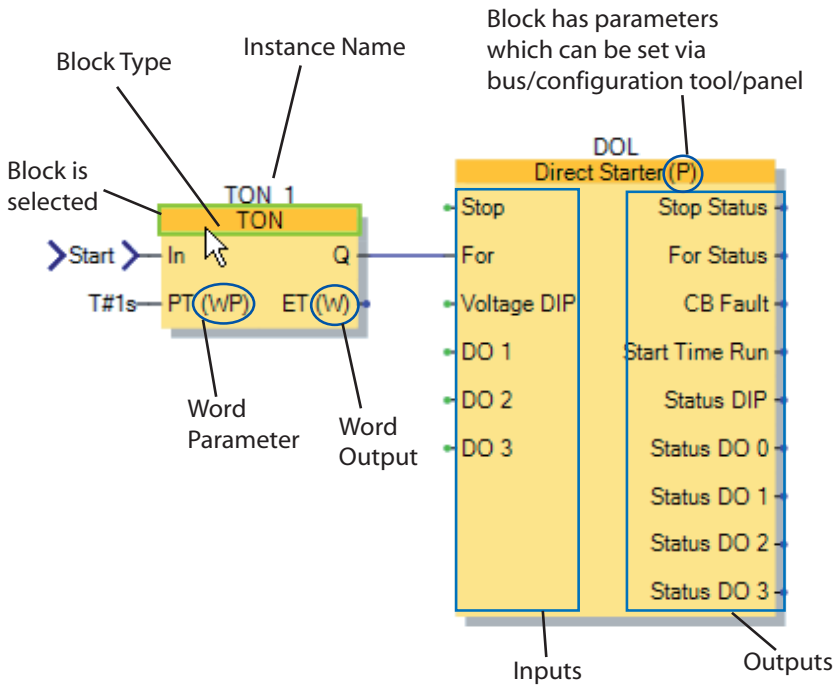
Parameters:
Mandatory terminals call for data supply via the signal flow line in order to enable the block to operate correctly. The T_{ON} block has a parameter which defines the delay time.

Parameterisation:
Not all function blocks must be parameterised. The need for parameterisation is indicated with a (P) in either of the function block names.

A (P) as postfix to the function block name (e.g. Direct Starter (P)) indicates that this block can be configured via the FIM, fieldbus or LCD panel. These parameters can be changed at any time later on. It is not necessary to open the application editor to change these parameters.

A (P) as postfix to an input terminal indicates that a parameter value has to be connected to this input. Take the delay time of an T_{on} function block as an example. The parameter value can only be changed in the application editor. It is not possible to change these parameters via the fieldbus or LCD panel.

Application elements



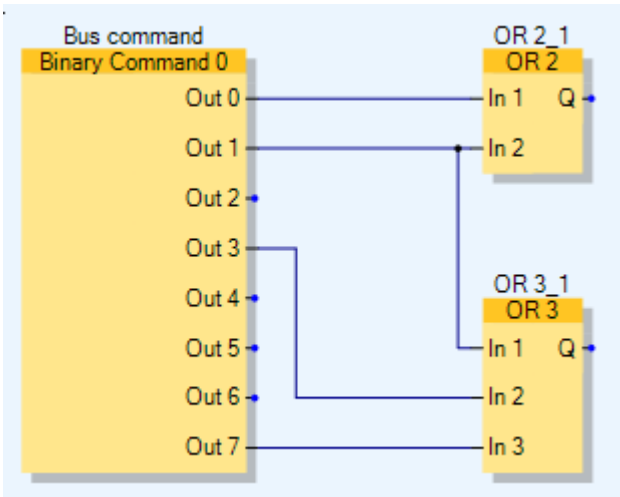
Connections and Lines

Connections can be made to parameter values and blocks. Connections are shown as horizontal or vertical lines. The editor has an auto router built in. You do not have to take care about the actual positioning of the connections. To connect an output with an input follow these steps:

1. Move the mouse over the output where you wish to start until the cursor changes from the pointer to a cross.
2. Then left click and move the mouse (a thin green line is attached) to an input port. If you are over the input port the thin green wire changes into a thicker green wire. Then left click again to connect the two pins.

The following rules apply:

- Only inputs can be connected with outputs. Outputs with outputs or inputs with inputs cannot be connected.
- It is possible to connect a boolean pin with a word pin. An error message is shown in the message window as soon as you start the checker.

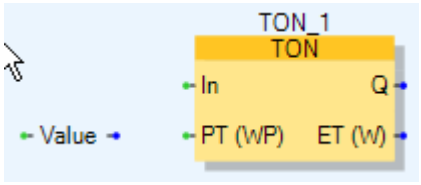


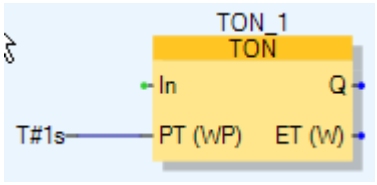
Application elements

Parameter Values

Parameter values can be placed anywhere on the worksheet, and are edited in a rectangle. They can be used to define parameters of function blocks or to define a constant value.

1. Place a value element on the worksheet (left diagram below)
2. Connect it to the parameter input (T_{ON} time in this case)
3. Double click on the value and type in the time
4. Press return (right figure below)
5. If the value is outside the permitted value range a message is shown in the message window if the application is checked (e.g. if you press Apply).





Parameters can have the following formats:

Time Constant

The supported time format follows the one used in the PLC world. It starts with the prefix 'T#' and then one or more of the following specifications are possible: h (hour), m (minute), s (second), ms (millisecond). In general an unspecified time input is internally set to zero (if not otherwise stated).

Examples:

- T#1h30m -> 1 hour, 30 min
- T200ms -> 200 milli seconds
- T1m20s100ms -> 1 minute, 20 seconds, 100 milli seconds

Boolean Constant

A boolean constant can be either true (logical 1) or false (logical 0).
Therefore a constant can be either '1' or '0'.
An unconnected input is internally set to zero (if not otherwise stated).

Word Constants

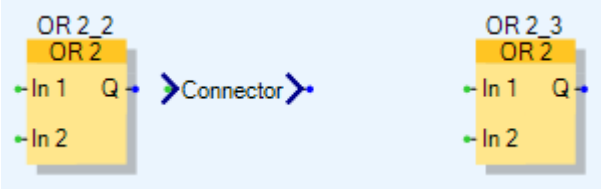
Word constants can have the value 0 ... 65535 respectively 0x0 ... 0xffff.
No negative values can be specified.
An unconnected input is internally set to zero (if not otherwise stated).

Application elements

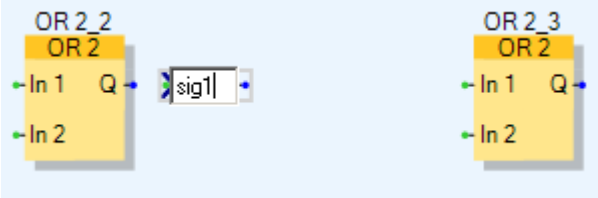
Connectors

Connectors are most useful for connecting two functions blocks which are far away from each other and would require many crossings. Two connectors with the same name are equal to a direct wire.
To use connectors follow these steps:

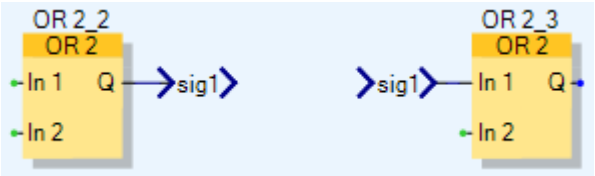
1. Drag a connector element from the library window and drop onto the worksheet



2. Double click on the connector and type in the name



3. If the left side of the connector is wired it becomes a sending connector. If the right side of the connector gets wired it becomes a receiving connector. To complete the connection add another connector and connect it with the target pin.



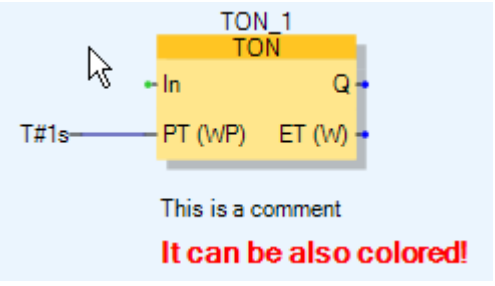
Left Open Pins

Left open pins are generally considered as logical 0 or 0. E.g. the output Q of a logical AND function block will never be logical 1 if one of its input pins is left open. If a parameter like a delay or counter value is left open it is considered as 0.

Comments

Comments can be used to describe the implemented logic for later reference. To add a comment drag it from the library window and drop it onto the worksheet. Double click on the comment field to change the text. Then press return.

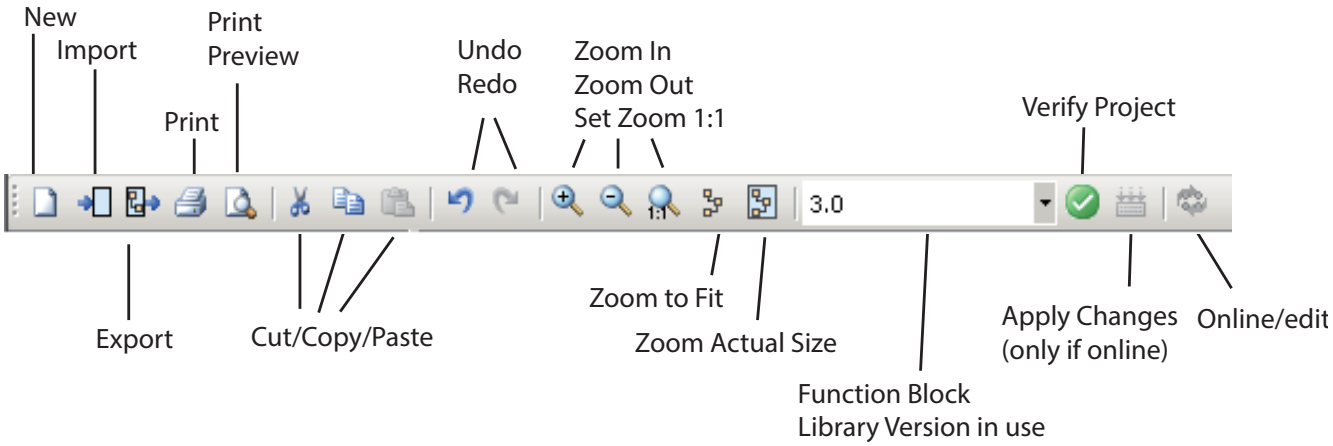
To change the text size or the font colour right click on the comment and select the Font context menu item.



Application elements

The Toolbar

The toolbar is at the top of the custom editor. It provides quick access to the most important functions.



Command	Description
New	Creates a new blank application. An existing application will be deleted.
Import	Import an application and replace the present one. This allows you to reuse existing applications from other projects.
Export	Exports the application to disk for later use.
Cut/Copy/Paste	Known Windows® functions to cut out, copy and paste selections (one or more elements)
Undo / Redo	Known Windows® functions to undo / redo changes. If you close the editor the Undo History is lost.
Zoom In (+)	Zoom in the worksheet to display more details.
Zoom Out (-)	Zoom out and display less details but a bigger part of the worksheet.
Zoom Actual Size 1:1	Zoom out to display the application in actual size.
Zoom to Fit	Zoom is set in a way that all function blocks of the application can be displayed. This function provides a full overview on the application.
Zoom Whole Diagram	Show the complete worksheet.
Function Block Library Version	The firmware of a UMC100 supports a specific set of function blocks. Newer firmware versions may have support for additional function blocks. The version selected here must match the one from the UMC firmware.
Verify Project	The checker checks that the application is consistent and can be executed from the UMC. e.g. it enforces that at least one control function block is used etc. If the checker reports any errors the application can be saved but the variables holding the application are set to zero. If a download to the UMC occurs now the UMC detects an invalid application and reports an error.
Apply	If you have made changes to the application in online mode press this button to download the logic into the UMC. The Apply button is only available in online mode.
Online	Toggle between online and edit mode. In online mode the signal status is shown

Application elements

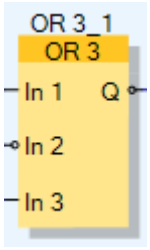
Editing a Function Block Application

The editor offers many functions to efficiently create applications. In the following section typical tasks are described and how they can be carried out.

Task	Method
Deleting blocks	Left click on a block and press the Delete button on your keyboard (see also Cut).
Moving blocks	Left click on a block and move the mouse around while keeping the key pressed.
Select multiple program elements	To select more than one element either a) press the Ctrl button on your keyboard and then click on the elements one by one that you wish to select. b) press the left mouse key on a place without an element and then move the mouse while keeping the mouse key pressed. All elements within the rectangle are selected.
Deselecting program elements	Click on an empty area on your worksheet to deselect a previous selection
Copy / Paste	To copy function blocks select a function block and press "Ctrl+C" on your keyboard. To paste the function blocks press "Ctrl+V". The function block names are automatically changed to make them unique. You can also use the icons in the toolbar.
Cut	To delete a selection press "Delete" on your keyboard or the toolbar icon.

Inverting Inputs or Outputs

The inputs and outputs of many function blocks in the group "Logic Functions" can be inverted. A negation is set or reset, for the selected terminal. To invert (negate) a terminal move the curser over the terminal until a cross appears. Then right click on the terminal and select "Negated". A negated terminal has a small circle added to it. In the example below the output Q and the input In2 are negated.



Note: Not all inputs and outputs can be inverted. Mainly the standard function blocks offer this possibility

Creating the first application

In this section you learn how to create an application step by step from scratch.

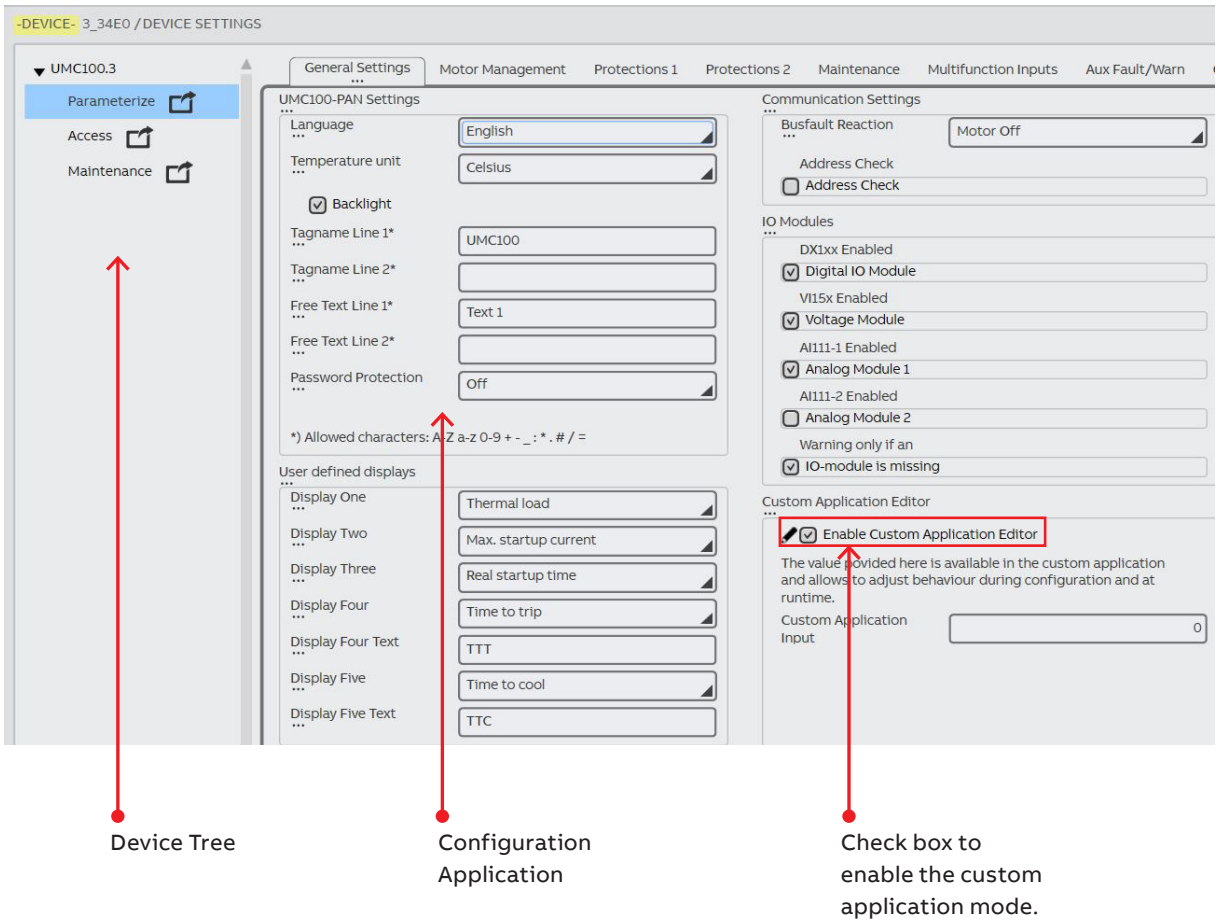
Preconditions:
FIM UMC Edition installed on your PC.

Communication interfaces depending on requirements
- Serial point-to-point USB cable connected to the UMC100-PAN control panel
- Profibus DP Communication interface UTP22-FBP.0 for Profibus and UMC100.3 with Profibus interface PDP32.0

Enabling the Custom Application Editor

After starting up FIM UMC Edition and selecting in Device Settings the Parametrize mode, the Custom Application Editor can be enabled by checking the parameter “Enable custom logic” on the tab “General Settings”. This brings up two additional tab cards, one with the application editor and a tab card for configuring the function block for triggering external warnings and trips.

Your display should look as shown below.



Device Tree

Configuration Application

Check box to enable the custom application mode.

Creating the first application

Sample Application

As a tutorial the following application shall be developed: The UMC100 controls a conveyor belt.
If the start command via DI4 is given a siren shall be switched on for 10 seconds. The siren is connected to the 24VDC output DO3. After the 10s delay the motor starts and the siren is switched off again. Monitoring shall be carried out via the LCD panel only. Resetting of faults shall always be possible via DI0 and the LCD panel. Stopping the motor via digital input DI5 shall be always possible. The simplified circuit diagram is shown below.

Drawing the Application

Clicking on the "Custom Application Editor" tab brings the editor to the foreground. By default no application is present and a blank workspace is shown. It is possible to load an existing application e.g. the standard direct starter from disk (-> Import) but for demonstration purposes we are starting from scratch.

Make sure version 3.0 of function block library is selected.
Only in this case all function blocks for the UMC100.3 are available.

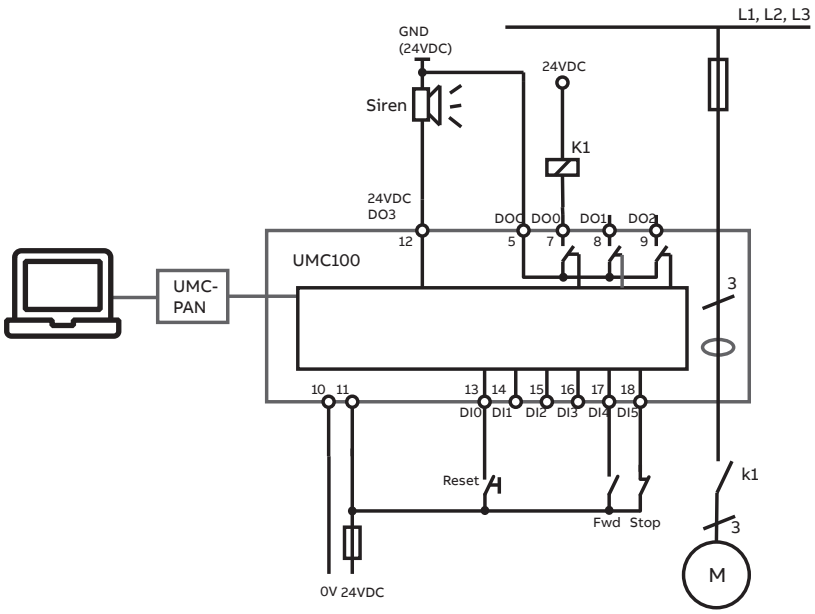
When creating a new application consider the following points:

- From which control places should motor control be possible?
  If control should be possible from more than one place (e.g. bus and digital inputs) it is highly recommended to use the Command Control function block.
- Which starter function is required?
  Select the right control function in the Motor Control tab and drag & drop the same function block as selected there on the workspace.
- What other functions are required?
  Is there a special function block already available or is it necessary to create the function with low level function blocks?

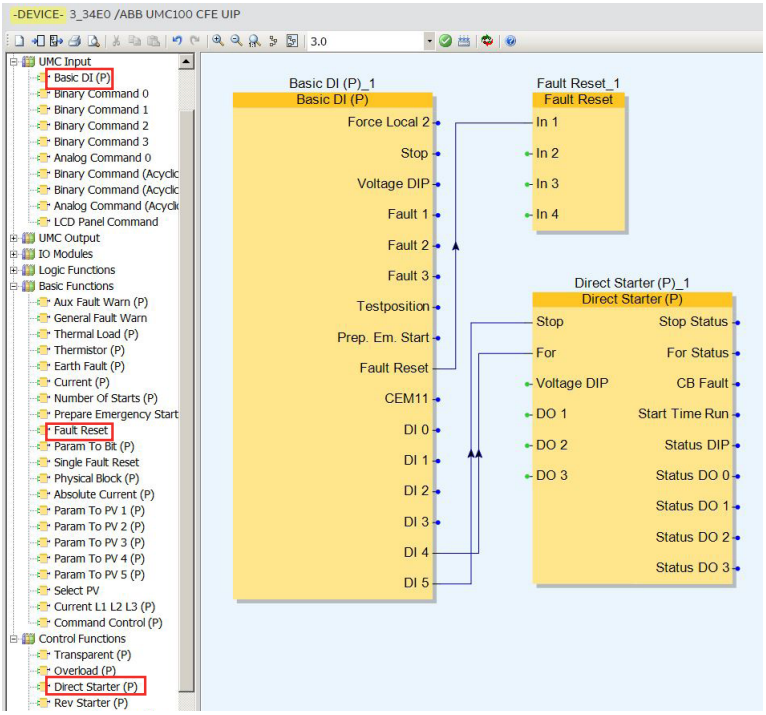
For this example the following function blocks are needed:

- DOL (in Control Functions)
- BasicDI (in UMC Inputs)
- Fault Reset (in Basic Functions)
- Some logic for the delayed start (logic added in a second step)

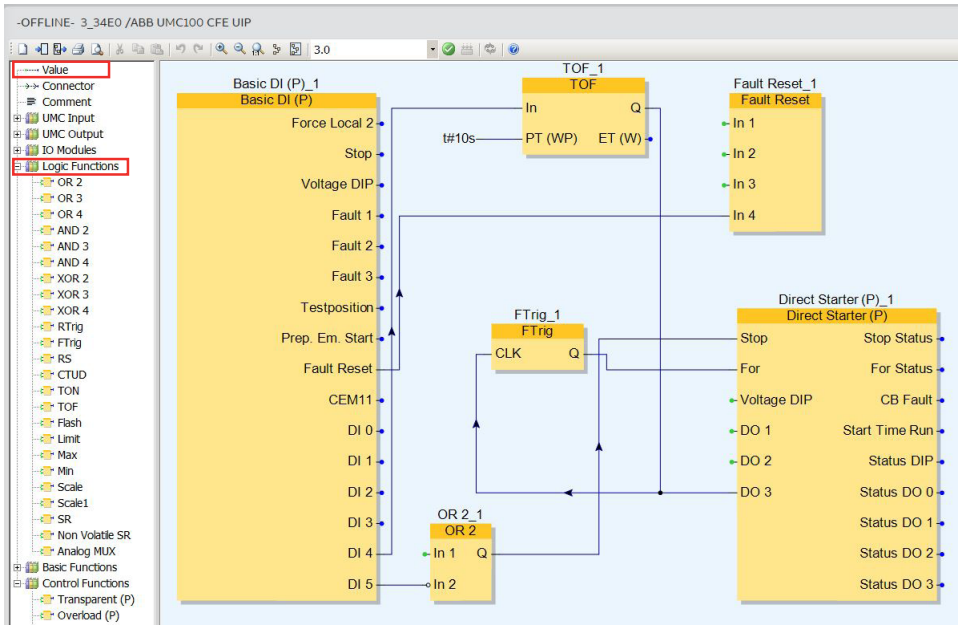
Drag these function blocks from the library and drop them onto the workspace. Then wire the inputs and outputs as shown below. To obtain a better overview about the circuit click “Zoom Out” until the display size fits your needs. The preliminary result is shown on the next page:



Creating the first application



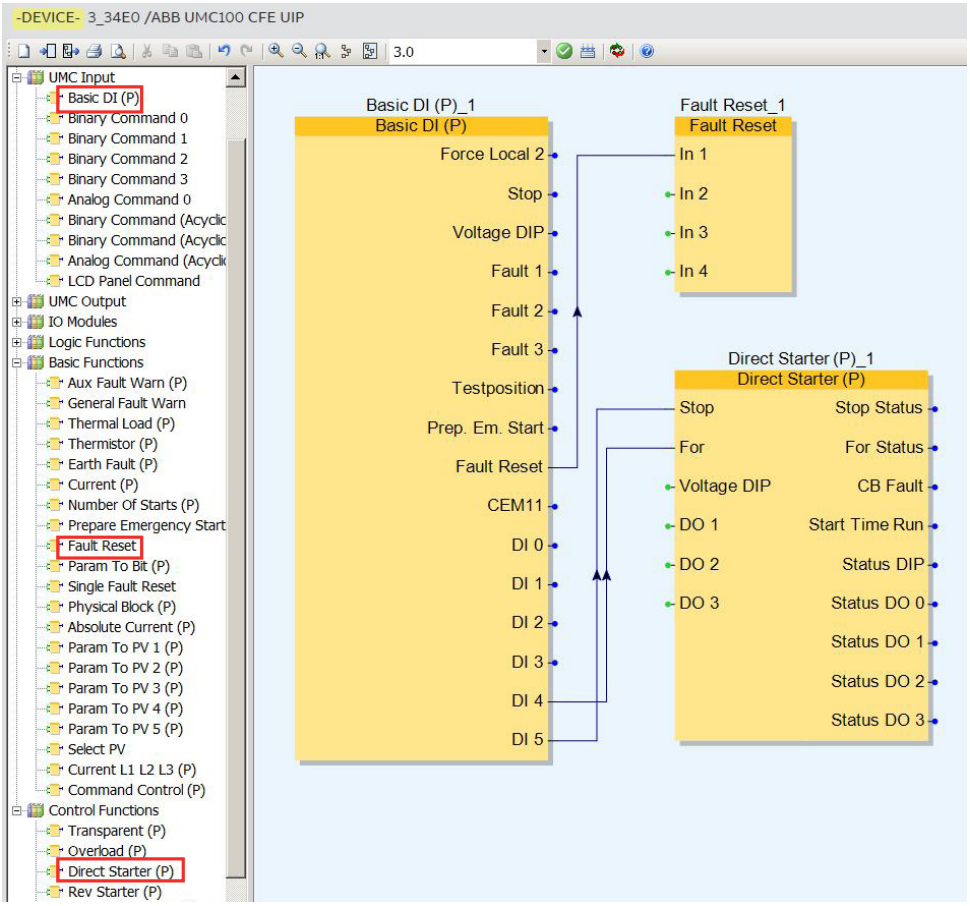
To check the application press the "Apply" button.
If there is any problem with the application an error message is displayed in the message window.
See section "Info and Error Messages" for a detailed list of all error messages. Fix the problem and press "Apply" again until no further errors are reported. For this simple application about 4% of the program memory is used. Save the project now. To complete the application some more logic is needed to realise the delay function. The complete application is shown in the next diagram:



The simple logic made of the TOF and FTrig blocks immediately sets the output DO3 following a start pulse at digital input DI4. After 10 seconds DO3 is switched off again.
The FTrig block detects the falling edge and sets its output to logical 1 for one cycle.
This triggers the DOL start input.
The DOL starter block contains all the logic for the starter function such as checkback supervision etc.

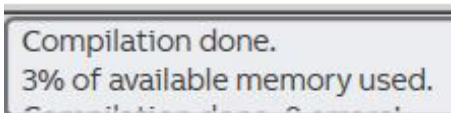
Creating the first application

Edit logic



The OR 2 block is used to invert the stop signal

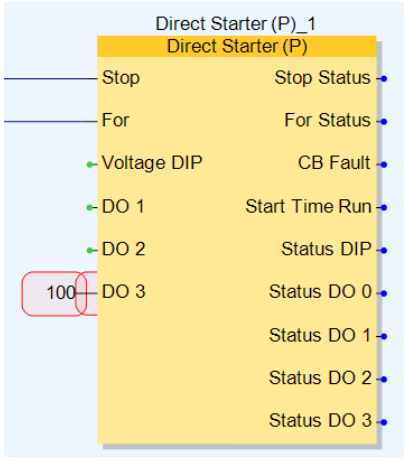
To check the application use the Verify Project button.



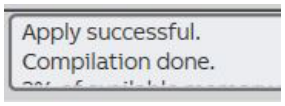
If there is any error in the application, it is shown in the editor and in the message window. In this example an analog value is connected to a binary input, which is not correct.

Creating the first application

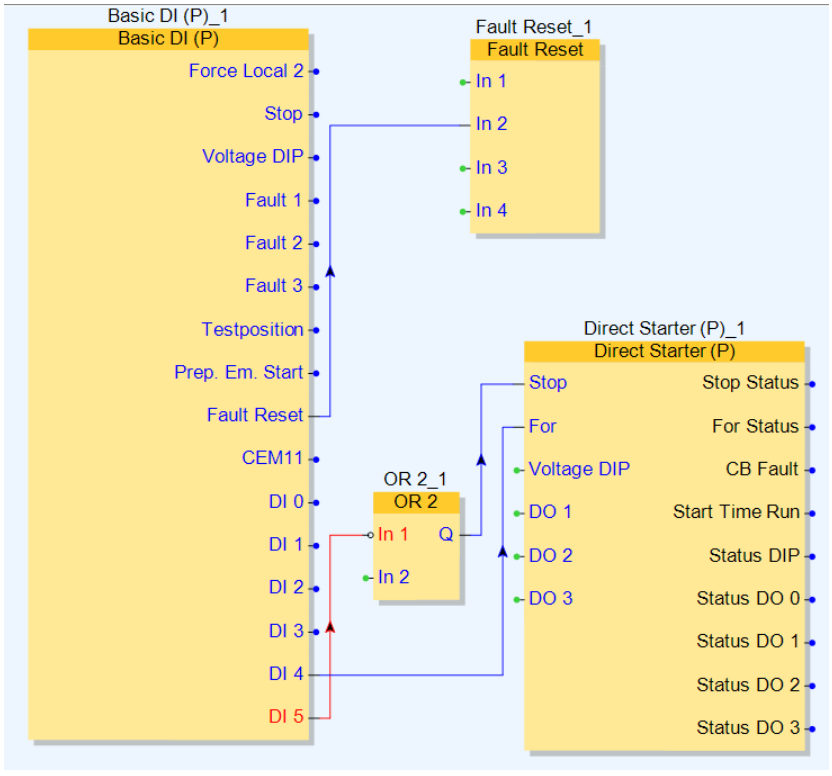
In this example an analog value is connected to a binary input, which is not correct.



To download the logic use the Apply button. The logic will only be downloaded if there is no error. The message window shows that apply and writing was successful. Download will take several seconds.



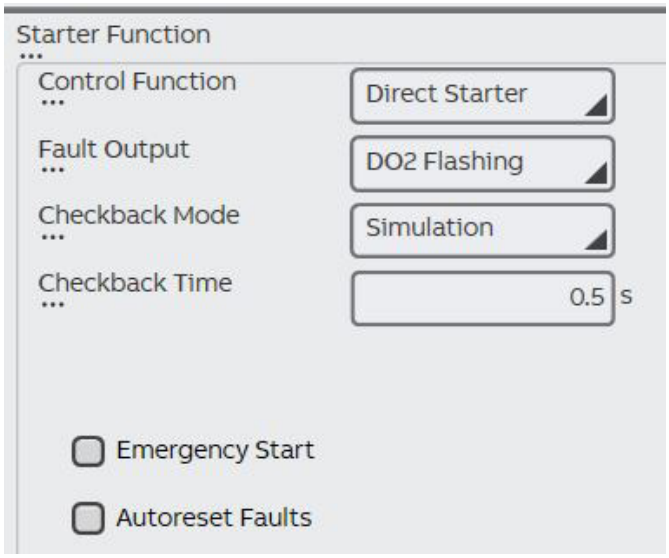
To show the online status, use the Online button. The screen is updated cyclically and shows a”1” signal in red color, “0” is shown in blue color. The message window shows, in which cyles the screen is updated.



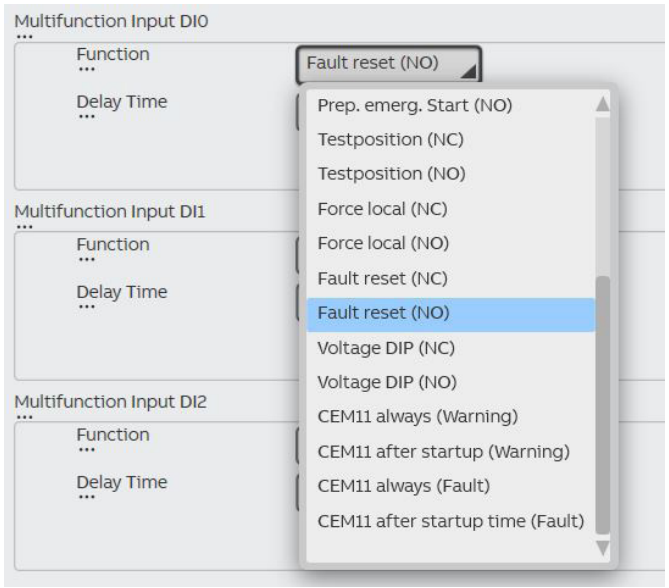
Creating the first application

To complete the starter according the drawing above, 2 parameter settings are required

- Control Function Direct Starter
- Fault Output DO3 Flashing
- Checkback Mode Simulation (this means to ignore checkback)



Multifunction input DI0 for Fault Reset (NO)



Creating the first application

—

Saving the logic

Take care when using the free trial version FIM Device Window Edition of the software. This version does not support saving projects. So export the custom application logic when you want to reuse it again. Next time it can be imported.

—

Summary

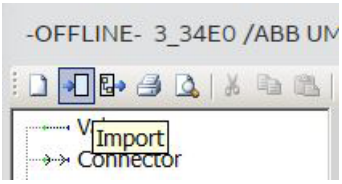
In this section a demo application was created to give you a quick start in using the Custom Application Editor. We started from scratch to show all steps in the custom logic development process. After finishing the development the application was checked and downloaded into the UMC. Then we showed you how to monitor the status of the application.

In most cases you will probably start with one of the predefined applications which are provided as templates and modify it according to your needs. This will be explained in the following section "Reusing Existing Solutions ".

Reusing existing solutions

To reuse existing solutions the following options exist:

- Import of a complete application which was previously exported from another project. This allows to reuse complete applications. Use the toolbar buttons to carry out the import and export. The imported application can now be adapted if necessary.



- Copy parts of an application and paste it into another application. This allows you to reuse parts of an application.
- Copy and paste is possible between two devices within one project and also between different projects. In the latter case both projects must be opened i.e. Asset Vision Basic must be started twice.

Make sure that the application or application parts you reuse are compatible with the UMC100 firmware version in use. See section "Backwards compatibilty" at the beginning of this manual for more information.

There is a reference application for each of the built-in control functions.

Example of a Direct Starter

The following diagram shows the complete application of a direct starter. On the left side of the diagram the function blocks are placed that deliver the inputs from the different control sources (bus, LCD panel, DI). In the middle the control place selection takes place. The start/stop signals are then connected to the control function block. On the right side of the diagram the function blocks responsible for monitoring (bus, LCD panel, DO) are placed

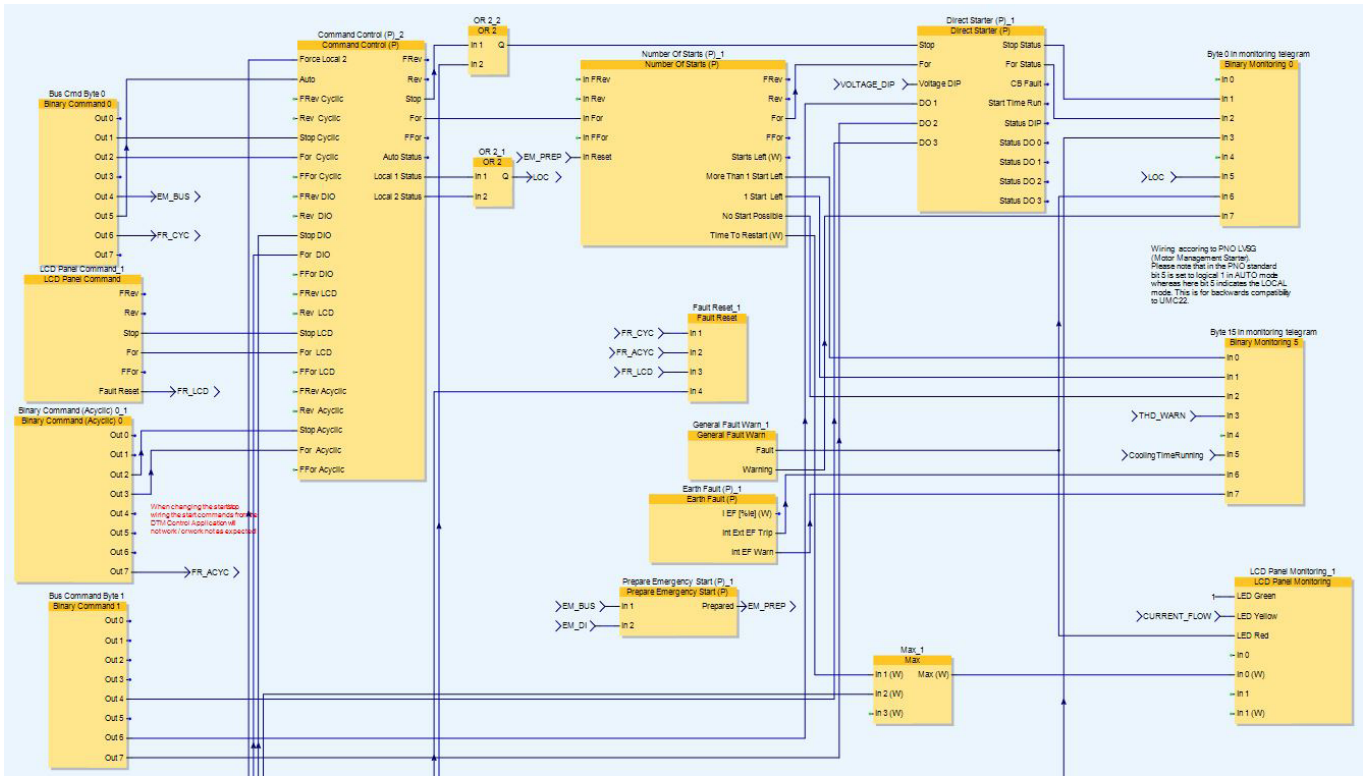
Reusing existing solutions

Source (Command) Function blocks

Control place selection

Starter – and other UMCfunctions

Monitoring blocks (Sinks)



Direct starter with full functionality as custom application. This application can be changed to fulfil specific requirements.

Function block reference

Overview

In this chapter all function blocks with their inputs, outputs and parameters are described. It is organised into the following subsections:

- UMC Input: Function blocks in this group represent inputs of the UMC (e.g. Digital Inputs, Command data from the bus).
- UMC Output: Function blocks in this group represent outputs of the UMC (e.g. Monitoring data)
- IO Modules: Function blocks in this group represent the inputs and outputs of the expansion modules. These function blocks can only be used if the related IO module is connected and enabled in the basic configuration. Otherwise the UMC will create an error "Module missing".
- Logic Functions: This group contains standard function blocks that represent simple boolean operations such as logic AND, OR, Inverter, Timers, Counters etc.
- Basic Functions: In this group basic UMC functions are represented as function blocks. The function blocks provide access to the basic UMC functions such as Fault Reset, Current Measurement, PTC Measurement ...
- Control Functions: In this group function blocks for the different motor starter functions are available.

Function block reference

UMC Input Function Blocks

The function blocks described here represent the different inputs of the UMC100 device.

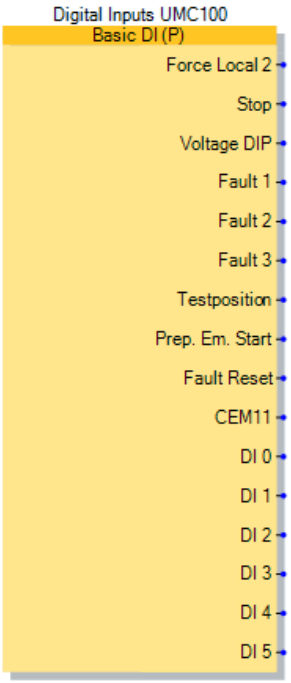
- Digital inputs DI0-DI5
- Cyclic fieldbus command telegramms

Word	Byte	Function Block
0	0	Binary Command 0
	1	Binary Command 1
1	2	Binary Command 2
	3	Binary Command 3
2	4	Analogue Command 0
	5	
3	6	
	7	

- Acyclic fieldbus command telegramms
- Commands from the LCD panel

Basic Input

This function bock represents the digital inputs of the UMC100. All six digital input signals are transparently available for further processing. The different functions of the three multifunction inputs DI0, DI1 or DI2 are also available. Each function has its separate output pin. To use the multifunction inputs the related parameters must be set accordingly. See the internal function diagram of the block on the next page.

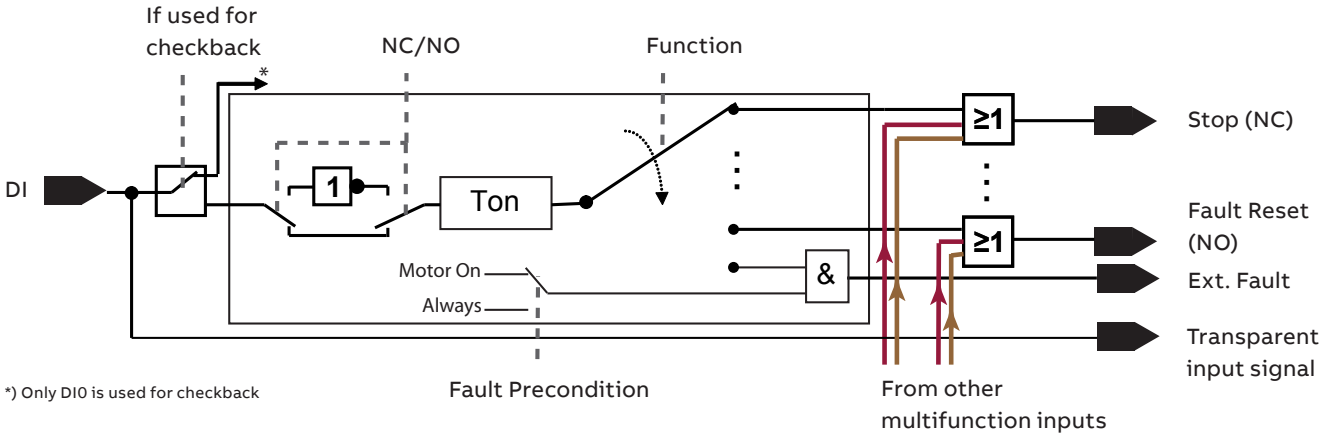
Function Block	

Function block reference

	Signal Name	Data Type
Inputs	-	-
Outputs (from top to bottom)	Force Local 2: Output is true if one of the multifunction inputs was configured to force the UMC into local mode 2 (see command control function block) and the corresponding input is set to true. To actually influence the mode this output must be connected to the 'Force Local 2' input of the CommandControl function block.	Boolean
	Stop: Output is true if one of the multifunction inputs was configured as stop input and the digital input is set. To actually stop the motor this output must be connected to the stop input of a starter block (e.g. DOL).	Boolean
	Voltage DIP: Output is true if one of the multifunction inputs was configured as voltage DIP input (e.g. from an external voltage monitor) and the digital input is set. To actually signal a voltage dip this output must be connected to the dip input of a starter block (e.g. DOL).	Boolean
	Fault 1 ... 3: Output is true if one of the multifunction inputs was configured as fault input and the corresponding digital input is set. It is not necessary to connect these outputs to any other block to create a fault. The fault signal is triggered internally.  After a trip the output becomes logical 0 immediately if the trip precondition has gone. If it is required to latch the output connect a flip-flop.	Boolean
	Testposition: Output is true if one of the multifunction inputs was configured as test input and the corresponding input is set. It is not necessary to connect this output to any other block to enable the testposition. The activation is carried out internally.	Boolean
	Prepare Emergency Start: Output is true if one of the multifunction inputs was configured to prepare an emergency start and the digital input is set. To actually prepare for an emergency start this output must be connected to one of the inputs of the function block "Prepare Emergency Start".	Boolean
	Fault Reset: Output is true if one of the multifunction inputs was configured as fault reset input and the corresponding input is set to true. To actually reset a fault the output must be connected to the function block "Fault Reset".	Boolean
	CEM11: Output is true if one of the multifunction inputs was configured as CEM11 input and an earth fault was detected. It is not necessary to connect the output to any other block. The earth-fault/warning signal is triggered internally.  After a trip the output becomes logical 0 immediately if the trip precondition has gone. If it is required to latch the output connect a flip-flop.	Boolean
	DI0 ... DI5: Transparent input signal of the related input.	Boolean
Parameters	The following parameters define the behaviour of the function block. For more information about the single parameters see the UMC100 manual. • Multifunction 0/1/2 • Multif. 0/1/2 delay • Multif. 0/1/2 autoreset (if ext. fault function) • Fault text Multif. 0/1/2 (if ext. fault function)	

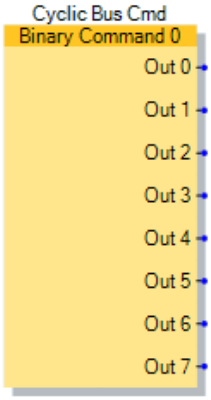
The block diagram of the multifunction inputs is shown on the next page.

Function block reference



Binary Command

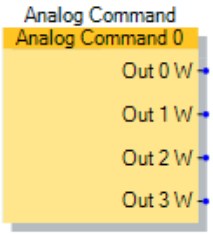
The function bocks "Binary Command 0 ...3" provide access to the command bytes 0 ... 3 cyclically sent via fieldbus (e.g. PROFIBUS) to the UMC.

Function Block	

	Signal Name	Data Type
Inputs	-	-
Outputs (from top to bottom)	Out 0 ... Out 7: The 8 bits of the command byte 0,1,2 or 3	Boolean
Parameters	-	

Analog Command 0

The function bocks "Analog Command 0" provide access to the two analog words cyclically sent via fieldbus (e.g. PROFIBUS) to the UMC.

Function Block	

	Signal Name	Data Type
Inputs	-	-
Outputs (from top to bottom)	Out 0 ... 3 (W): First to forth command word from the fieldbus	Word
Parameters	-	

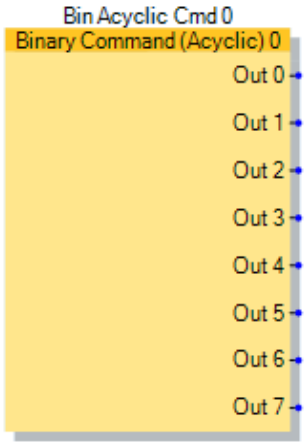
Function block reference

Binary Command Acyclic

The function blocks "Binary Command Acyclic 0 ...1" provide access to the command bytes 0 ... 1 acyclically sent via fieldbus (e.g. PROFIBUS) to the UMC.
Acyclic commands are usually sent from a service tool upon user request.

	The function block BinAcyclicCmd0 behaves slightly differently to BinAcyclicCmd1. By default it is used to receive start/stop commands sent from the FIM. Therefore commands received at the block are only available for one application cycle at its outputs. Then the outputs are cleared automatically. The mapping from the command buttons to the outputs is as follows: Bit 0: Fast Reverse Bit 1: Reverse Bit 2: Stop Bit 3: Forward Bit 4: Fast Forward Bit 5: Reserved Bit 6: Reserved Bit 7: Fault Reset

The command bytes can be sent to Slot 4/Index 54 Byte 0 (BinAcyclicCmd0) and Byte 1 (BinAcyclicCmd01). The total data block length is 8 Bytes whereas the last 6 bytes are unused.

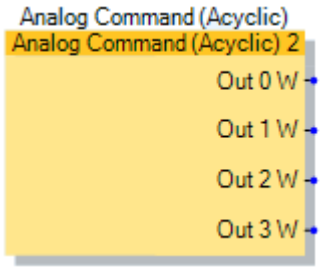
Function Block	

	Signal Name	Data Type
Inputs	-	-
Outputs (from top to bottom)	Out 0 ... Out 7: The 8 bits of the command byte 0 or 1	Boolean
Parameters	-	

Function block reference

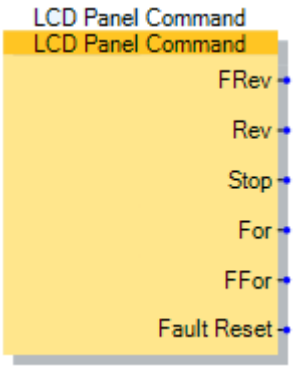
Analogue Command Acyclic 0

The function bocks "Analog Command Acyclic 0" provide access to the four analog words acyclically sent via fieldbus (e.g. PROFIBUS) to the UMC. Acyclic commands are usually sent from a service tool. The command words can be sent to Slot 4/ Index 55 word 0-3. The total data block length is 8 bytes (4x2 bytes).

Function Block		
	Signal Name	**Data Type**
Inputs	-	-
Outputs (from top to bottom)	Out 0 (W): First word from bus.	Word
	Out 1 (W): Second word from bus.	Word
	Out 2 (W): Third word from bus.	Word
	Out 3 (W): Forth word from bus.	Word
Parameters	-	

LCD Panel Command

This function block provides access to the commands given from the LCD panel.

Function Block		
	Signal Name	**Data Type**
Inputs	-	-
Outputs (from top to bottom)	FRev: Fast reverse start command	Boolean
	Rev: Reverse start command	Boolean
	Stop: Stop command	Boolean
	For: Forward start command	Boolean
	FFor: Fast forward start command	Boolean
	Fault Reset: Fault reset command	Boolean
Parameters	-	

Function block reference

UMC Output Function Blocks

The function blocks described in this section represent the different outputs of the UMC100 device.

- Cyclic fieldbus monitoring telegram

Word	Byte	Function Block
0	0	Binary Monitoring 0
	1	Binary Monitoring 1
1	2	Analog Monitoring 0
	3	
2	4	
	5	
3	6	
	7	
4	8	
	9	
5	10	
	11	
6	12	Binary Monitoring 2
	13	Binary Monitoring 3
7	14	Binary Monitoring 4
	15	Binary Monitoring 5

Mapping between data in the cyclic monitoring telegramm and the function blocks.

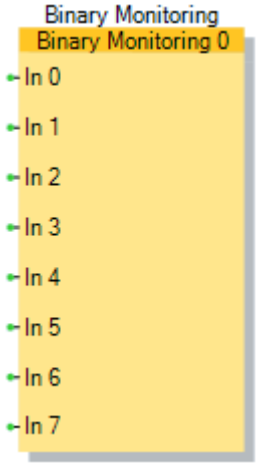
- Acyclic fieldbus monitoring telegram
- LEDs on the LCD panel

The digital outputs of the UMC100 are only accessible via the Control Function blocks which are described in section "Motor Management" of the UMC100 manual.

Binary Monitoring

The function bocks "Binary Monitoring 0 ... 5" provide access to the monitoring bytes 0 ... 5 cyclically sent via fieldbus (e.g. PROFIBUS). Monitoring data is sent from the UMC to the fieldbus master (e.g. a PLC).

Function Block

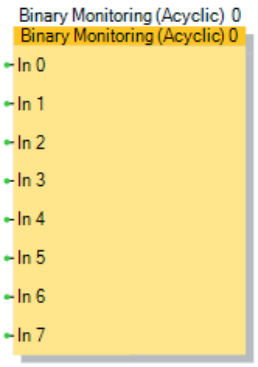


	Signal Name	Data Type
Inputs (from top to bottom)	In 0 ... In7: The 8 bits of the monitoring byte 0, 1, 2, 3, 4 or 5	-
Outputs	-	-
Parameters	-	

Function block reference

Binary Monitoring Acyclic

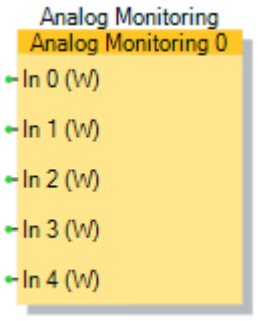
The function bocks "Binary Monitoring Acyclic 0 ... 1" provide access to the monitoring bytes 0 ... 1 acyclically sent via fieldbus (e.g. PROFIBUS). Monitoring data is sent from the UMC to the fieldbus master (e.g. a PLC) only upon request.
The monitoring bytes can be read at slot 4 / index 7 (Binary Monitoring Acyclic 0) and index 8 (Binary Monitoring Acyclic 1). 8 Bytes must be read whereas byte 0 represents the monitored byte.

Function Block	

	Signal Name	Data Type
Inputs (from top to bottom)	In 0 ... In7: The 8 bits of the monitoring byte 0, 1, 2, 3, 4 or 5	-
Outputs	-	-
Parameters	-	

Analogue Monitoring

The function blocks "Analog Monitoring" provide access to the monitoring words 0 ... 4 which are sent cyclically via fieldbus (e.g. PROFIBUS) to the master. Monitoring data is sent from the UMC to the Fieldbus master (e.g. a PLC) on request only.

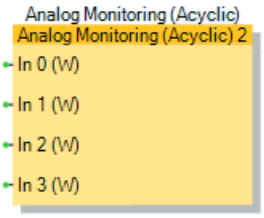
Function Block	

	Signal Name	Data Type
Inputs (from top to bottom)	In 0 (W) ... In 4 (W): The five monitoring words.	Word
Outputs	-	-
Parameters	-	

Function block reference

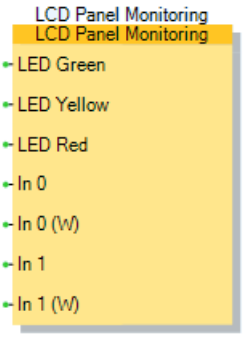
Analog Monitoring Acyclic

The function bock "Analog Monitoring Acyclic 2" allows to set the four monitoring words 0 ... 3. They are sent via fieldbus (e.g. PROFIBUS) from the UMC to the master only upon request.

Function Block		
	Signal Name	**Data Type**
Inputs (from top to bottom)	In 0 (W) ... In 3 (W): The four monitoring words.	Word
Outputs	-	-
Parameters	-	

LCD Panel Monitoring

This function block allows you to control the LEDs on the LCD panel. The LCD panel also displays process data at the top level menu (e.g. the motor current). Beside the predefined data the values in two menu points can be changed by the user with the help of this function block. Each menu point allows to display either a boolean or a word value.

Function Block		
	Signal Name	**Data Type**
Inputs (from top to bottom)	LED Green / Yellow / Red: If true the corresponding LED on the panel is on (glowing)	Boolean
	In 0: Boolean value that shall be displayed on the LCD panel at menu point "User Display 4". It is possible to define a text shown on the display to explain the value using menu point " User display 4 text"	Boolean
	In 0 (W): Word value that shall be displayed on the LCD panel at menu point "User Display 4". It is possible to define a text shown on the display to explain the value using menu point " User display 4 text"	Word
	In 1: Boolean value that shall be displayed on the LCD panel at menu point "User Display 5". It is possible to define a text shown on the display to explain the value using menu point " User display 5 text"	Boolean
	In 1 (W): Word value that shall be displayed on the LCD panel at menu point "User Display 5". It is possible to define a text shown on the display to explain the value using menu point " User display 5 text"	Word
Outputs	-	-
Parameters	-	

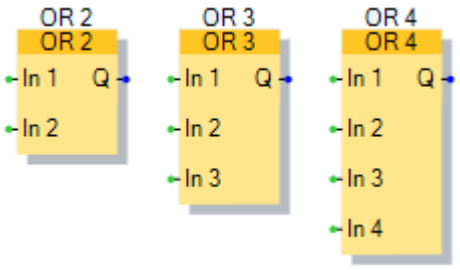
Function block reference

Logic Functions

This section describes standard logic blocks which are similar to the function defined in the IEC61131 standard.

Logic OR

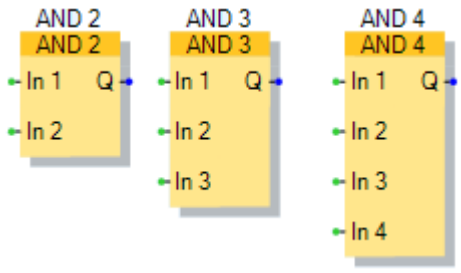
This function performs bit-by-bit OR interconnection of the inputs and passes on the result to the output. The max. number of inputs is 4.

Function Block	

	Signal Name	Data Type
Inputs (from top to bottom)	In 1 ... In 4: Input signals	Boolean
Outputs	Q: Output	Boolean
Parameters	-	

Logic AND

This function performs a bit-by-bit AND interconnection of the inputs and passes on the result to the output. The max. number of inputs is 4.

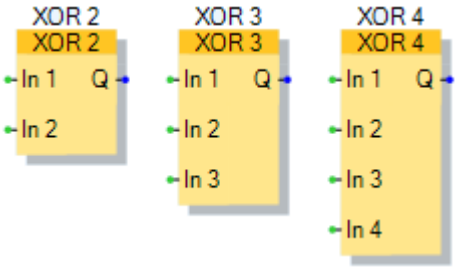
Function Block	

	Signal Name	Data Type
Inputs (from top to bottom)	In 1 ... In 4: Input signals	Boolean
Outputs	Q: Output	Boolean
Parameters	-	

Function block reference

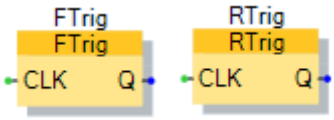
XOR

This function performs bit-by-bit exclusive OR interconnection of the inputs and passes on the result to the output. The max. number of inputs is 4.

Function Block		
	Signal Name	**Data Type**
Inputs (from top to bottom)	In 1 ... In 4: Input signals	Boolean
Outputs	Q: Output	Boolean
Parameters	-	

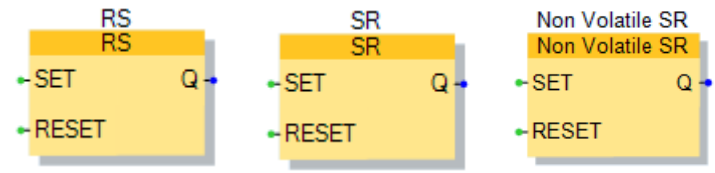
FTrig, RTrig

If a positive edge appears at input CLK for the function RTrig, output Q is set to logic-1 signal. If a negative edge appears at input CLK for the function FTrig, output Q is set to logic-1 signal. Output Q remains set to logic-1 signal for one cycle.

Function Block		
	Signal Name	**Data Type**
Input	CLK: Input	Boolean
Output	Q: Output	Boolean
Parameters	-	

RS, SR (Flip-Flop)

The flip-flop is used for storing logical binary states. A logical 1 signal at the SET input sets output Q, while a logical 1 signal at input RESET sets output Q back again. If both inputs are set the RESET input is dominant in the RS function block whereas the SET input is dominant for the SR function block. The 'Non Volatile SR' flip-flop stores its state also during a power cycle. It can be used to trigger an action based on an event before the power cycle.

Function Block		
	Signal Name	**Data Type**
Inputs	SET: Set input	Boolean
	RESET: Reset input	Boolean
Output	Q: Output	Boolean
Parameters	-	

Function block reference

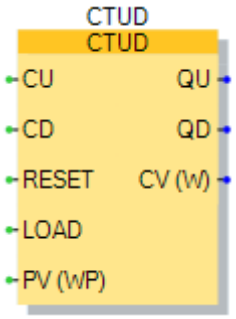
Counter Up/Down

This block can be used to monitor discrete processes or quantity measurements with pulse generators. An internal, unsigned 16-bit counter contains the balance of the up and down pulses (CU, CD) since the last reset.

The overflow limits, both positive and zero, are monitored. If the counter value exceeds PV the overflow output QU is set. If the counter value reaches 0 QD is set.

After a LOAD pulse, the configurable basic value PV is accepted as the initial counter value. After a reset RES the value 0 (zero) is set as the initial value. Reset is dominant.

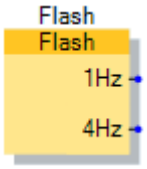
A CU and CD signal at the same time does not change the counter.

Function Block		
	Signal Name	**Data Type**
Inputs (from top to down)	CU: A positive edge increments the counter by one if PV is reached.	Boolean
	CD: A positive edge decrements the counter by one. If zero is reached the timer is not further decremented and the QD output is set.	Boolean
	RES: A logical 1 resets the counter to zero.	Boolean
	LOAD: A positive edge loads the counter to the value PV	Boolean
	PV (WP): The initial counter value (0x0 ... 0xFFFF) when using LOAD.	Word
Outputs (from top to down)	QU: Output set to logical 1 if the counter reaches the predefined value PV.	Boolean
	QD: Output set to logical 1 if the counter reaches 0 (zero).	Boolean
	CV(W): Actual timer value	Word
Parameters	-	

Function block reference

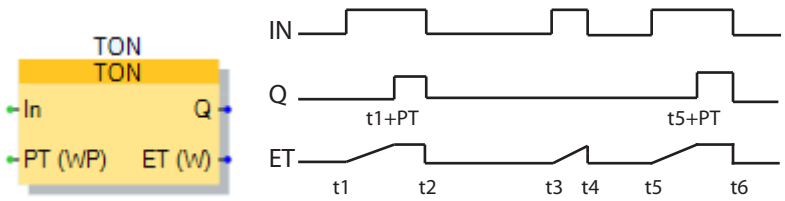
Flash

This block creates a 1Hz and a 4Hz waveform at its outputs. It can be used to switch an output on/off (e.g. a fault output) cyclically.

Function Block		
	Signal Name	**Data Type**
Inputs	-	-
Output	1Hz: Output toggles between logical 1 and logical 0 with 1 Hz	Boolean
	4Hz: Output toggles between logical 1 and logical 0 with 4 Hz	Boolean
Parameters	-	

TON (Timer, Switch-On Delay)

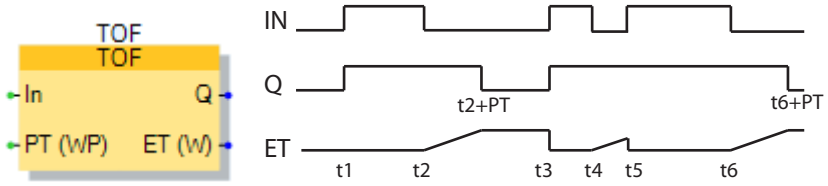
The function block Timer On Delay implements a switch-on delay.

Function Block		
	Signal Name	**Data Type**
Inputs	In, PT: In and PT are input variables of the Boolean and Time types respectively. If In is logical 0, Q is logical 0 and ET is 0. As soon as In becomes logical 1, the time will begin to be counted in multiples of 100 milliseconds in ET until its value is equal to PT. It will then remain constant. t#1H15M22S 1 hour, 15 minutes, 22 seconds t#1H 1hour t#10s500ms 10.5 seconds Internally the delay time is represented as multiples of 100ms. The max. possible delay time is 1h49m13s500ms. If PT is left open the delay time is set to zero. In online mode the output ET (W) shows the remaining delay time in multiples of 100ms.	Boolean, Time
Output	Q and ET are output variables of the boolean and Time types respectively. Q is logical 1 when IN is logical 1 and ET is equal to PT. Otherwise it is logical 0.	Boolean, Time
Parameters	-	

Function block reference

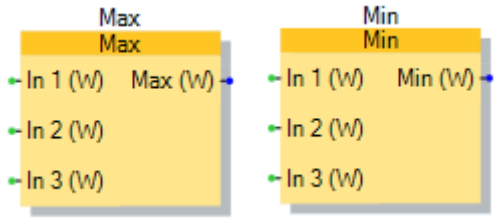
TOF (Timer, Switch-Off Delay)

The function block TOF implements a turn-off delay.

Function Block		
	Signal Name	**Data Type**
Inputs	IN and PT are input variables type BOOL respectively TIME. If IN is logical 1, the outputs are logical 1 respectively logical 0. As soon as IN becomes logical 0, at ET the time will begin to be counted in multiples of 100ms until its value is equal to PT. It will then remain constant. t#1H15M22S 1 hour, 15 minutes, 22 seconds t#1H 1hour t#10s500ms 10.5 seconds Internally the delay time is represented as multiples of 100ms. The max. possible delay time is 1h49m13s500ms. If PT is left open the delay time is set to zero. In online mode the output ET (W) shows the remaining delay time in multiples of 100ms.	Boolean, Time
Output	Q is logical 0 when IN is logical 0 and ET equal PT. Otherwise it is logical 1.	Boolean, Time
Parameters	-	

Max / Min

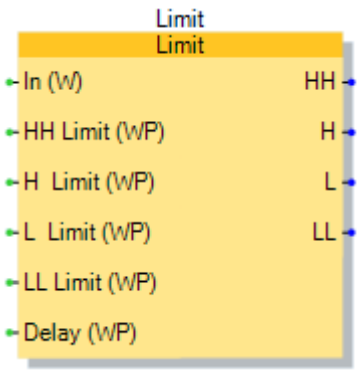
The greatest / smallest respective input value is passed on to the output.

Function Block		
	Signal Name	**Data Type**
Inputs (from top to bottom)	In1 (W): Analogue input 1	Word
	In2 (W): Analogue input 2	Word
	In3 (W): Analogue input 3	Word
Output	Max (W) / Min (W): The greatest or the smallest of the three inputs.	Word
Parameters	-	

Function block reference

Limit

This block reads an analog input and checks it against four thresholds. If a threshold is exceeded the corresponding output is set to true.

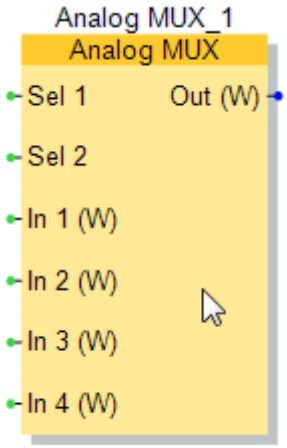
Function Block		
	Signal Name	**Data Type**
Inputs (from top to down)	In (W): Analog input	Word
	HH Limit (WP): If the analog signal In is above this value the output HH ist set to logical 1	Word
	H Limit (WP): If the analog signal In is above this value the output H ist set to logical 1	Word
	L Limit (WP): If the analog signal In is below this value the output L is set to logical 1	Word
	LL Limit (WP): If the analog signal In is below this value the output LL is set to logical 1	Word
	Delay (WP): The analog signal must remain above the threshold during the delay period at least to activate the corresponding output. Internally the delay time is represented as multiples of 100ms. The max. possible delay time is 1h49m13s500ms. If PT is left open the delay time is set to zero.	Word
Output	HH: High-High Output	Boolean
	H: High Output	Boolean
	L: Low Output	Boolean
	LL: Low-Low Output	Boolean
Parameters	-	

The HH Limit, H Limit, L Limit and LL Limit inputs must be specified in descending order. Otherwise the outputs HH, H, L and LL will remain low all the time.

Function block reference

Analog Mux (Multiplexer)

Allows to switch between multiple input words based on 2 digital inputs in a binary manner

Function Block	

	Signal Name	Data Type
Inputs	Sel1/Sel2: Selects which input is connected to the output. Selection is done in BCD manner: 00=In1, 01=In2, 10=In3, 11=In4	Boolean
	In1/2/3/4 (W): Analogue input word	Word
Output	Out (W): Analog value selected from one of the inputs. By default In1 is used.	Word
Parameters	-	

Function block reference

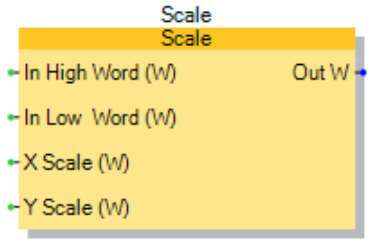
Scale

This function block allows to scale a word (16 Bit) value or a double word (32 Bit) value provided as two words.

Calculation rule: Out = (In * X Scale) / Y Scale

If the calculated result is > 65535 the output will be limited to 65535. If the results is < 0 the output is limited to 0. If a word should be scaled it has to be connected to "In Low Word". The "In High Word" input can be left open in this case (internally set to 0).

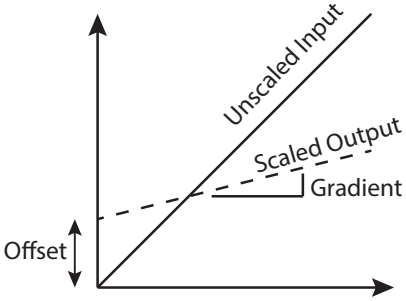
For an example see section "DX1xx Output".

Function Block		
	Signal Name	**Data Type**
Inputs (from top to down)	In High Word (W): Upper 16 Bits of the double word value	World
	In Low Word (W): Lower 16 Bits of the double word value	Word
	X Scale: Nominator	Word
	Y Scale: Denominator	Word
Output	Out (W): Results of the scaling operation	Word
Parameters	-	

Function block reference

Scale1

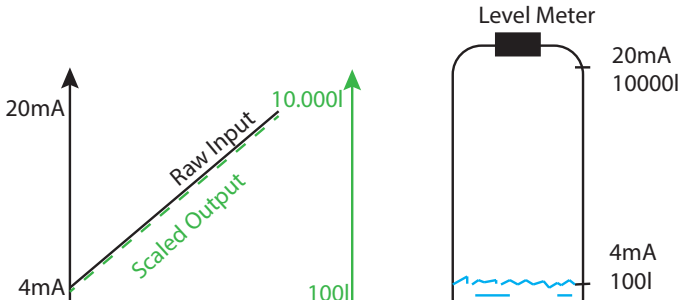
This function block allows to scale a word (16 Bit).
Calculation rule: Out = In * X Scale / Y Scale + P Offset – N Offset



Gradient = X Scale / Y Scale

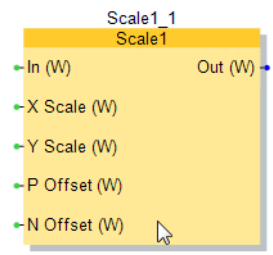
If the calculated result is > 65535 the output will be limited to 65535.
If the results is < 0 the output is limited to 0.

Example: User wants to show fill level in liters on the LCD display. Depending on fill level the motor shall be started to refill. Focus here is only on scaling. The level sensor was wired to an analog input and configured to 4-20mA.



Instrument input	Analog Value in UMC	Scaled to Liters
4mA	0	100l
...	...	...
20mA	27648	10000

Solution: Connect In(W) to the analog input channel, set X-scale to 9900, Y-scale to 27648, P Offset to 100, N Offset to 0.
Out = Input * 9900 liters / 27648 + 100 liters

Function Block	

	Signal Name	Data Type
Inputs (from top to down)	In High Word (W): 16 Bit value to scale	World
	X Scale: Nominator of Gradient	Word
	Y Scale: Denominator of Gradient	Word
	P Offset: Positive Offset	Word
	N Offfset: Negative Offset	Word
Output	Out (W): Results of the scaling operation	Word
Parameters	-	

Function block reference

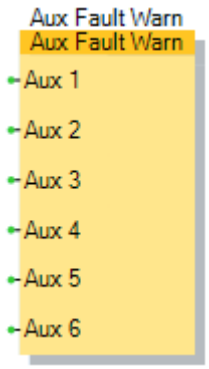
Basic Functions

Function blocks described in this section represent basic UMC functions as function blocks to provide access to data or to enable the predefined behaviour to be changed.

AuxFaultWarn

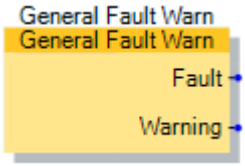
This function block provides six digital inputs which can be configured as either warning or trip inputs. For each input it is possible to

- - configure a delay time
- - define a message which is displayed on the LCD panel in the event of a fault / warning
- - optionally set/reset a fault automatically if the fault input is logical 0 again

Function Block		
	Signal Name	**Data Type**
	Aux 1 ... Aux 6: Inputs one to six	Boolean
	-	-
Parameters	The following parameters define the behaviour of the function block. For more information about the single parameters see the UMC100 manual. Aux. Input 1-6 Reaction Aux. Input 1-6 Mode Aux. Input 1-6 Delay Aux. Input 1-6 Message Text Line 1/2	

General Fault Warn

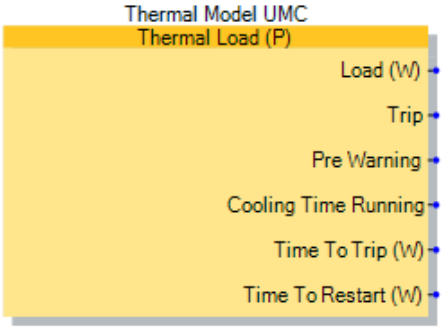
Provides two outputs which indicate that there is a fault (Trip) or warning detected by the UMC. These signals can be used to control other functions e.g. a relay output.

Function Block		
	Signal Name	**Data Type**
Inputs	-	-
Outputs	Fault: Indicates that there is a Fault. The source could be diverse e.g. a thermal overload trip, a bus fault ...	Boolean
	Warning: Indicates that there is a Warning. The source could be diverse e.g. the motor current is above a defined threshold..	Boolean
Parameters	-	

Function block reference

Thermal Load

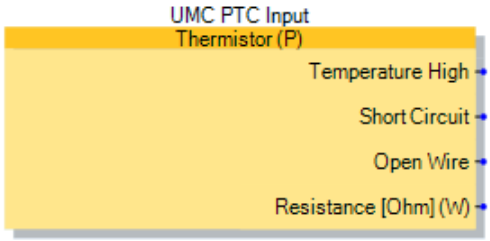
This function block provides access to values coming from the motor model. They can be used for further processing or to connect them to any output block (e.g. bus monitoring).

Function Block		
	Signal Name	**Data Type**
Inputs	-	-
Outputs (from top to bottom)	Load (W): Thermal load of the motor as a percentage. 100% is the trip level.	Word
	Trip: Logical 1 after a thermal overload trip as long as the trip was not acknowledged.	Boolean
	Pre Warn: The output is logical 1 if the thermal load is above a user-defined threshold	Boolean
	Cooling Time Running: Logical 1 as long as the cooling time after a thermal overload trip is running.	Boolean
	Time to Trip (W): Time in seconds until the motor gets tripped because of thermal overload.	Word
	Time to Restart (W): Time in seconds the user has to wait until the UMC accepts a start command after a thermal overload trip.	Word
Parameters	The following parameters define the behaviour of the function block. For more information about the single parameters see the UMC100 manual (Section: Protection Parameters -> General Protection Parameters). Thermal Load Warnlevel Cooling Model Cooling Time Restart Level %	

Function block reference

Thermistor

This function block provides access to the PTC input signals.

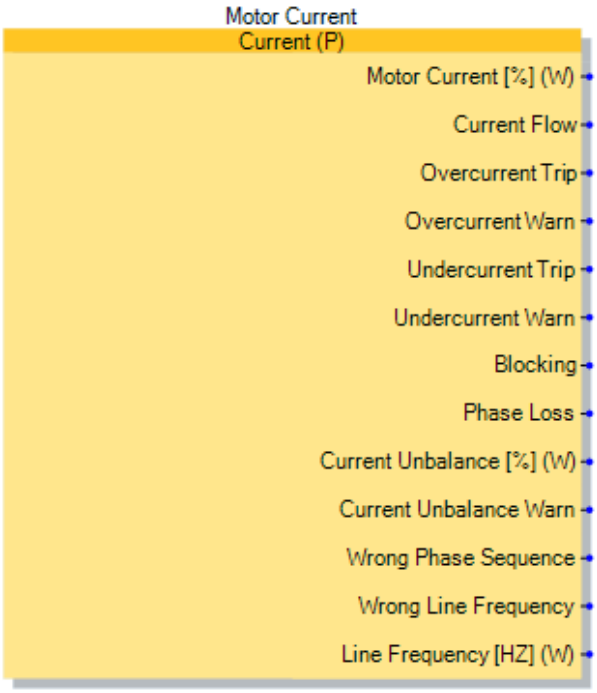
Function Block		
	Signal Name	**Data Type**
Inputs	-	
Outputs (from top to bottom)	Short-Circuit: The output is logical 1 if the UMC has detected a short-circuit at the PTC inputs.	Boolean
	Open Wire: The output is logical 1 if the UMC has detected a cable break at the PTC inputs.	Boolean
	Temperature High: The output is logical 1 if the PTC resistance is above the hot threshold.	Boolean
	Resistance: Value measured from the PTC input in ohms.	Word
Parameters	The following parameter defines the behaviour of the function block. For more information about the single parameters see the UMC100 manual. PTC	

Function block reference

Current

This function block provides access to the signals of the motor current measurement unit of the UMC.

Function Block

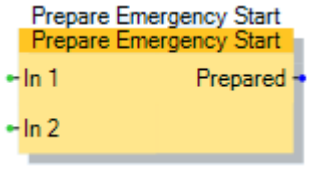


	Signal Name	Data Type
Inputs	-	-
Outputs (from top to bottom)	Motor Current: Actual current as a percentage of I_e. A value of 0 corresponds to 0%. A value of 800 to 800%.	Word
	Undercurrent Trip / Warn: The output is set to logical 1 if the actual current is below the user-defined undercurrent trip level / warn level.	Boolean
	Overcurrent Trip / Warn: The output is set to logical 1 if the actual current is above the user defined overcurrent trip / warn level.	Boolean
	Blocking (locked rotor): The output is set to logical 1 if a blocking fault was detected.	Boolean
	Phase Loss: The output is set to logical 1 if a phase loss was detected.	Boolean
	Current Imbalance [%](W): Actual current imbalance in % (0% means no imbalance).	Word
	Current Unbalance Trip / Warn: The motor current unbalance is above the user defined warning / trip level.	Boolean
	Wrong Phase Sequence: The phases are in the wrong order. Normally the order shall be L1/ L2/L3.	Boolean.
	Wrong Line Frequency: The line frequency is outside the specified limits.	Boolean
	Line Frequency [Hz] (W): The acutal line frequency in Hz. A value of XXX means 50Hz. A value of YYY meams 60Hz.	Word
	Current Flow: The output is set to logical 1 as soon as current flow is detected.	Bool
Parameters	See chapter "Motor Protection Functions" of the UMC100 technical manual.	

Function block reference

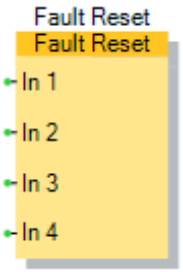
Prepare Emergency Start

Before an emergency start is possible a release signal has to be sent. This function block allows you to set the release signal.

Function Block		
	Signal Name	**Data Type**
Inputs	In1, In2: Inputs to enable an emergency start.	Boolean
Output	Prepared: If the output is logical 1 an emergency start can be performed.	Boolean
Parameters	The following parameter defines the behaviour of the function block. For more information about the single parameters see the UMC100 manual. Emergency Start	

Fault Reset

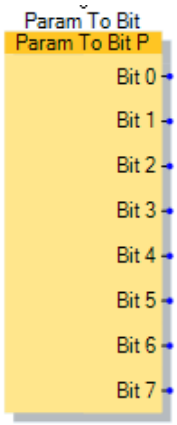
The fault reset function block allows the acknowledgement of faults.

Function Block		
	Signal Name	**Data Type**
Inputs	In1, In2, In3, In4: Reset a fault if input is logical 1.	Boolean
Outtput	-	-
Parameters	-	

Function block reference

Param To Bit

Normal parameters that can be set via fieldbus allow you to configure the predefined functions of the UMC.
There is one parameter byte which has no predefined function but can be used to adjust the function of its own function block application. This can be used for example to enable or disable a part of the user-defined logic.

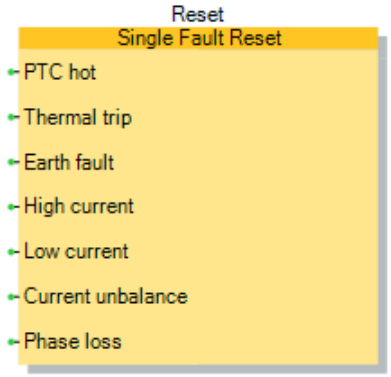
Function Block	

	Signal Name	Data Type
Inputs	-	-
Output	Bit 0 - Bit 7: Bits of the byte specified in the custom application parameter	Boolean
Parameters	The following parameter defines the output of the function block. For more information about the single parameters see the UMC100 manual. Custom App Parameter	

Function block reference

Resetting a Single Fault

This function block allows to reset only a particular fault.

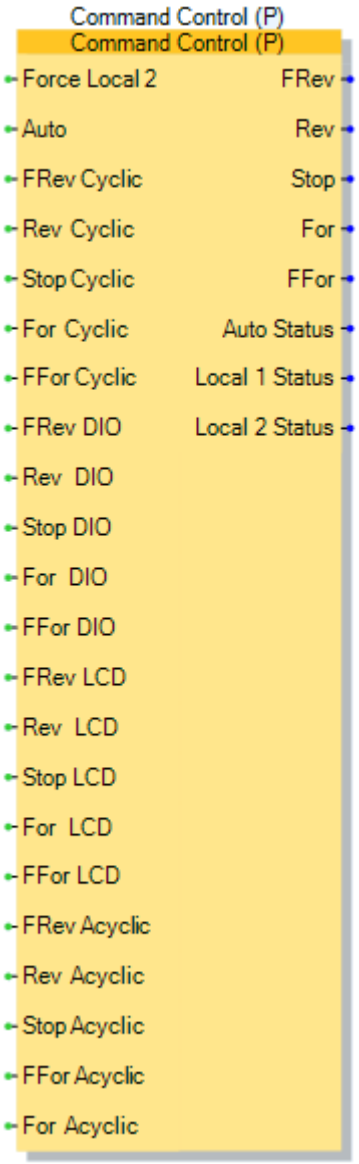
Function Block		
	Signal Name	**Data Type**
Inputs	PTC hot: Allows to acknowledge a PTC trip	Boolean
	Thermal trip: Allows to acknowledge a thermal overload trip	
	Earth fault: Allows to acknowledge an earth fault trip	
	High current / Low current: Allows to acknowledge a current level trip	
	Current imbalance: Allows to acknowledge an imbalance fault reported from the voltage or current measurement subsystem.	
	Phase loss: Allows to acknowledge a phase loss fault reported from the voltage or current measurement subsystem.	
Output	-	-
Parameters	-	

Function block reference

Command Control

The Command Control function block provides the complete logic to select between three independent modes (Auto, Local 1, Local 2) based on the input signals (bus fault (internal signal), Force Local 2 and Auto). For each mode the user can independently define from which control source a start or stop shall be possible. In addition it handles the device reaction in the event of a bus fault.

If there are special requirements for the start/stop logic the function block can be left out. In this case all the related parameters have no meaning anymore. This includes the bus fault function too!

Function Block	

Function block reference

<table>
<tr><th></th><th>Signal Name</th><th>Data Type</th></tr>
<tr><td rowspan="6">Inputs (from top to bottom)</td><td>Force Local 2, Auto:
These two inputs allow you to select one out of the three modes "Local 1", "Local 2" and "Auto". The table below shows the selected mode based on the three input signals.
<table>
<tr><td>Busfault (internal signal)</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Force Local 2</td><td>x</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>Auto</td><td>x</td><td>0</td><td>x</td><td>1</td></tr>
<tr><td>Resulting Mode</td><td>Local 1</td><td>Local 1</td><td>Local 2</td><td>Auto</td></tr>
</table></td><td>Boolean</td></tr>
<tr><td>FRevCyclic (Fast Reverse Cyclic), RevCyclic (Reverse Cyclic), StopCyclic, ForCyclic (Forward Cyclic), FForCyclic (Fast Forward Cyclic):
These command inputs are intended to be connected to a cyclic fieldbus telegram (command telegram) function block - e.g. the Command byte 0 and Command byte 1 blocks.</td><td>Boolean</td></tr>
<tr><td>FRev DIO (Fast Reverse), Rev DIO (Reverse Cyclic), Stop DIO, For DIO (Forward), FFor DIO (Fast Forward):
These command inputs are intended to be connected to the digital inputs of the UMC e.g. the "Basic DI" function block.</td><td>Boolean</td></tr>
<tr><td>FRev LCD (Fast Reverse), Rev LCD (Reverse Cyclic), Stop LCD, For LCD (Forward), FFor LCD (Fast Forward):
These command inputs are intended to be connected to the LCD panel function block.</td><td>Boolean</td></tr>
<tr><td>FRev Acyclic (Fast Reverse), Rev Acyclic (Reverse Cyclic), Stop Acyclic, For Acyclic (Forward), FFor Acyclic (Fast Forward):
These command inputs are intended to be connected to one of the acyclic command function blocks.</td><td>Boolean</td></tr>
<tr><td rowspan="4">Outputs (from top to botton)</td><td>FRev (Fast Reverse), Rev (Reverse), Stop, For (Forward), FFor (Fast Forward):
These outputs are command outputs and usually connected to a starter block (e.g. DOL).</td><td>Boolean</td></tr>
<tr><td>Auto Status: Indication that the auto mode is active for command selection.</td><td>Boolean</td></tr>
<tr><td>Local 1 Status: Indication that the Local 1 mode is active for command selection.</td><td>Boolean</td></tr>
<tr><td>Local 2 Status: Indication that the Local 2 mode is active for command selection.</td><td>Boolean</td></tr>
<tr><td>Parameters</td><td colspan="2">See sections "Configuring the Motor Management Functions" and " Configuring the Fieldbus Communication-> Defining the Bus Fault Reaction" in the UMC100 manual.</td></tr>
</table>

The outputs FRev, Rev, For and FFor are set if there is a start signal at the related inputs. If the connected starter function block cannot start for any reason (e.g. cooling time still running, lockout time running) the above mentioned outputs are reset automatically. A new start signal must be issued to trigger another start.

Function block reference

Limit the Number of Starts

This function block can limit the number of starts issued in a given time window.

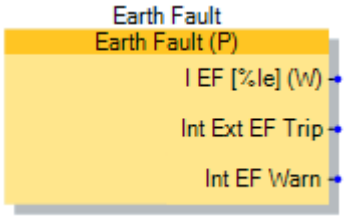
Function Block	

	Signal Name	Data Type
Inputs	In FRev, In Rev, In For, In FFor: Start commands	Boolean
	In Reset: Resets the function block to the inital state. I.e. the number of counted starts in the time window is set to zero and the outputs updated accordingly.	Boolean
Output	FRev, Rev, For, FFor: Start commands passed the block if a start is allowed.	Boolean
	Starts Left (W): Counter indicating how many starts are remaining in the time window.	Word
	More Than 1 Start Left: More than one start is possible	Boolean
	1 Start Left: Only one start is possible anymore	Boolean
	No Start Possible: No start command is accepted anymore. The user has to wait the "Time To Restart" time until a new start is accepted.	Boolean
	Time To Restart: Time in seconds until a new start command is accepted.	Word
Parameters	Num Starts Allowed Num Starts Window Num Starts Pause Num Starts Overrun Num Starts Prewarn	

Function block reference

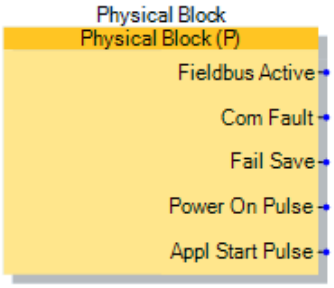
Earth Fault

This block provides access to the earth fault protection status and the calculated earth fault current.

Function Block		
	Signal Name	**Data Type**
Inputs	-	-
Output	I EF [I_e] (W): Earth fault current in % of the set current I_e	Word
	Int Ext EF Trip: Trip signal from the internal earth fault calculation or from the external CEM11 device.	Boolean
	Int EF Warn: The internally calculated earth fault current has exceeded the warn level.	Boolean
Parameters	Earth Flt Warn Level Earth Flt Warn Delay	

Physical Block

This block provides access to different UMC internal status signals.

Function Block		
	Signal Name	**Data Type**
Inputs	-	-
Output	Fieldbus Active: If high the UMC exchanges cylic I/O telegrams with the FieldbusPlug.	Boolean
	Com Fault: If high one of the following problems is present: • Connection to FieldbusPlug lost • Supervision of the master in the FieldbusPlug was triggered • Connection between the FieldbusPlug and the fieldbus master interrupted.	Boolean
	Fail Safe: If high the Profibus master has left the state "Data Exchange" or the PLC is in stop mode.	Boolean
	Power On Pulse: A 500ms pulse signals the power on of the UMC. This signal can be used to trigger actions that should only run once after UMC start.	Boolean
	Appl Start Pulse: A 500ms pulse signals a restart of the application (e.g. after downloading it with the tool). This output can be used to trigger actions that should run after each application start.	Boolean
Parameters	-	

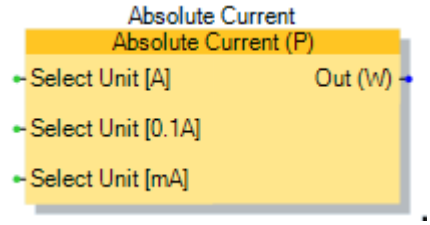
Function block reference

Absolute Current

The absolute motor current can be transferred as analog value via fieldbus. With Param to PV it can be selected which value, the average value of 3 phases or from any phase.
The default unit is [A] without digits.

With the Absolute Current block the unit can be selected.
If no input is selected, [A] is taken
If several inputs are set, the highest unit will be taken [A, 0.1A, mA}
The output does not need to be connected, unless you want to use this value in the custom application editor.

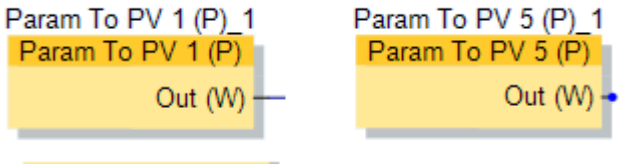
Take care to select the unit in a way that the output does not overrun (e.g. during overload). If the output is 65535 an overrun has occurred. In this case select the next higher unit.

Function Block		
	Signal Name	**Data Type**
Inputs	Selected Unit [1 A] or [0.1 A] for [1 mA]. Examples: Unit is set to A. A value at Out of 5 means 5A Unit is set to 0.1A: A value at Out of 5 means 500mA (5*0.1A) Unit is set to mA: A value at Out of 250 means 250mA (250*0.001A)	Boolean
Output	Out (W): The absolute motor current in the selected unit.	Word
Parameters	-	

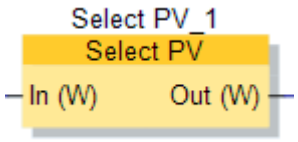
Function block reference

Parameter to Process Value

With the help of the following function blocks it is possible to select which analog value is cyclically transmitted just by setting parameters e.g. in the PROFIBUS GSD file. By default all five analog monitoring words are connected to these function blocks. No programming is required anymore to change an analog value.

Function Block		
	Signal Name	**Data Type**
Inputs	-	-
Output	Out (W): The selected analog value	Word
Parameters	Param To PV 1... 5	

With the help of the following function block it is possible to dynamically change the transmitted analog value.

Function Block		
	Signal Name	**Data Type**
Inputs	In (W): The value selects one of the internally available analog values and makes it available on the output. See UMC100.3 manual parameter "Param To PV1" to get a full mapping list. Extract: 0 Current [%] 1 Thermal load [%] 2 Max Value (TTT and Time To Restart from FB Number of starts) 3 TTC 4 Scaled Active Power (scaled with power factor) 5 Max Motor Current At Startup [%] 6 Real Motor Startup Time ...	Word
Output	Out (W): The selected analog value	Word
Parameters	-	

Function block reference

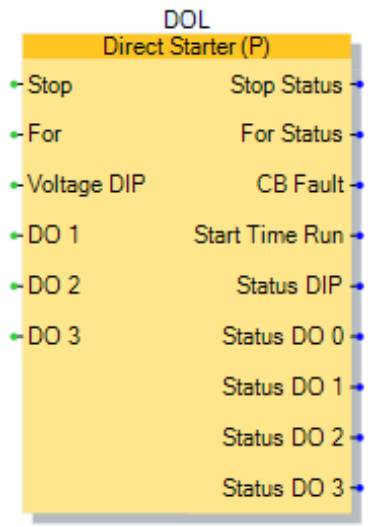
Control Functions

A control function block is available for the different type of starters supported by UMC. The function blocks realise the complete start / stop logic as well as the timing and checkback supervision of the contactor(s).
Depending on the control function some of the digital outputs have predefined functions. Therefore the digital outputs are not accessible directly but only via these function blocks. If an output is needed by the function block itself it is not available for the application.

The function blocks also handle the fault output which can be configured in steady state or flashing mode.

Direct Online Starter (DOL)

Use this function in a feeder that requires a motor to start/stop in one direction of rotation.

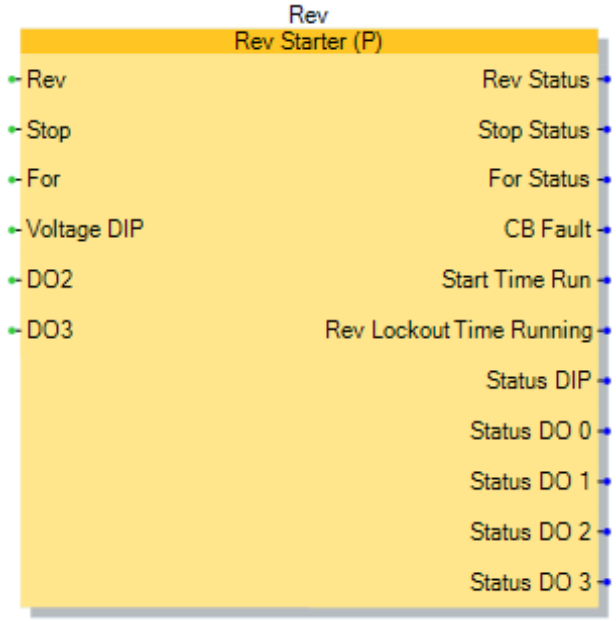
Function Block	

	Signal Name	Data Type
Inputs (from top to bottom)	Stop: If logical 1 the motor will be stopped and cannot be started until the input changes to logical 0 level.	Boolean
	For: Start motor	Boolean
	Voltage DIP: Signals a voltage DIP situation which is triggered by an external device (e.g. central voltage monitor).	Boolean
	DO1 ... DO3: Inputs not used from the starter. Depending on the fault output configuration also DO2/DO3 is controlled from the function block.	Boolean
Output (from top to bottom)	Stop Status: If the motor has been stopped this output is set to active 1	Boolean
	For Status (Forward): If the motor is running the output is set to active 1	Boolean
	CB Fault: A checkback fault has occurred.	Boolean
	Start Time Run: During motor start this output is set to logical 1.	Boolean
	Status DIP: Signals that the voltage DIP function is active. See UMC100 manual for more information about voltage DIP.	Boolean
	Status DO0 ... DI3: Mirror of the digital outputs of the UMC.	Boolean
Parameters	See section "Control Function Direct Starter" in the UMC100 manual.	

Function block reference

Reversing Starter (REV)

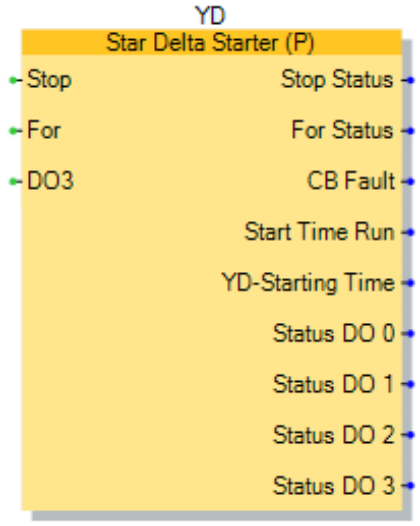
Use this function in a feeder that requires a motor to start/stop in two directions of rotation (forward / backwards).

Function Block		
	Signal Name	**Data Type**
Inputs (from top to bottom)	Rev: Start motor in a reverse direction	Boolean
	Stop: If logical 1 the motor will be stopped and cannot be started until the input changes to logical 0 level.	Boolean
	For: Start motor in a forward direction	Boolean
	Voltage DIP: Signals a voltage DIP situation which is triggered by an external device (e.g. central voltage monitor).	Boolean
	DO2 / DO3: Inputs not used from the starter. Depending on the fault output configuration also DO2/DO3 is controlled from the function block.	Boolean
Output (from top to bottom)	Rev Status (Reverse): If the motor is running in reverse direction the output is set to active 1	Boolean
	Stop Status: If the motor has been stopped this output is set to active 1	Boolean
	For Status (Forward): If the motor is running in forward direction the output is set to active 1	Boolean
	CB Fault: A checkback fault has occurred.	Boolean
	Start Time Run: During motor start this output is set to logical 1.	Boolean
	Status DIP: Signals that the voltage DIP function is active. See UMC100 manual for more information about voltage DIP.	Boolean
	Rev Lockout Time Running: When locical-1 the lockout time is running and a start signal is ignored.	Boolean
	Status DO0 ... DI3: Mirror of the digital outputs of the UMC.	Boolean
Parameters	See section "Control Function Reversing Starter" in the UMC100 manual.	

Function block reference

Star-Delta Starter

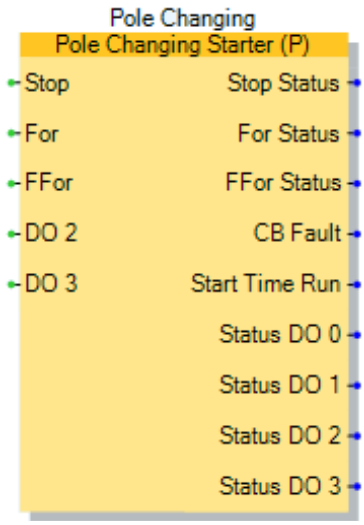
Use this function in a feeder that requires the motor to start in star-delta mode.

Function Block		
	Signal Name	**Data Type**
Inputs (from top to bottom)	Stop: If logical 1 the motor will be stopped and cannot be started until the input changes to logical 0 level.	Boolean
	For: Start motor	Boolean
	DO3: Output not used from the starter. Depending on the fault output configuration also DO3 is controlled by the function block.	Boolean
Output (from top to bottom)	Stop Status: If the motor has been stopped this output is set to active 1	Boolean
	For Status (Forward): If the motor is running the output is set to active 1	Boolean
	CB Fault: A checkback fault has occurred.	Boolean
	YD Starting Time: This output is logical 1 in star mode and logical 0 in delta mode.	Boolean
	Start Time Run: During motor start this output is set to logical 1.	Boolean
	Status DO0 ... DI3: Mirror of the digital outputs of the UMC.	Boolean
Parameters	See section "Control Function Star-Delta Starter" in the UMC100 manual.	

Function block reference

Pole-Changing Starter

Use this function in a feeder that requires the motor to start in star-delta mode.

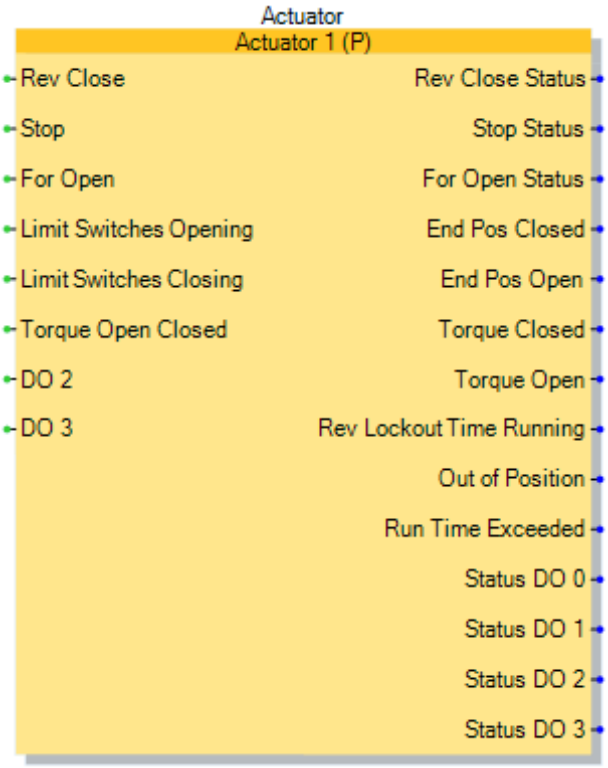
Function Block		
	Signal Name	**Data Type**
Inputs (from top to bottom)	Stop: If logical 1 the motor will be stopped and cannot be started until the input changes to logical 0 level.	Boolean
	For: Start motor in slow speed mode.	Boolean
	FFor (Fast Forward): Start motor in fast speed mode.	
	DO2... DO3: Inputs not used from the starter. Depending on the fault output configuration also DO2/DO3 is controlled by the function block.	Boolean
Output (from top to bottom)	Stop Status: If the motor has been stopped this output is set to active 1.	Boolean
	For Status (Forward): If the motor is running at slow speed the output is set to active 1.	Boolean
	FFor Status (Fast Forward): If the motor is running at fast speed the output is set to active 1.	Boolean
	CB Fault: A checkback fault has occurred.	Boolean
	Start Time Run: During motor start this output is set to logical 1.	Boolean
	Status DO0 ... DI3: Mirror of the digital outputs of the UMC.	Boolean
Parameters	See section "Control Function Pole Changing Starter" in the UMC100 manual.	

Function block reference

Actuator 1- 4

Use this function in a feeder that requires an open/close valve to be controlled. Select the actuator block that fits your requirements regarding limit and torque inputs.

Function Block

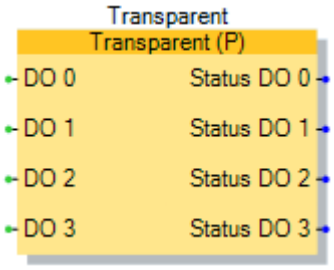


	Signal Name	Data Type
Inputs (from top to bottom)	Rev (Close): Start motor in a reverse direction..	Boolean
	Stop: If logical 1 the motor will be stopped and cannot be started until the input changes to logical 0 level.	Boolean
	For (Open): Start motor in a forward direction.	Boolean
	Limit Switches Opening / Closing: Inputs to signal the status of the limit switches.	Boolean
	Torque Open Closed: Input signalling the status of the torque switches.	Boolean
	DO2... DO3: Inputs not used from the starter. Depending on the fault output configuration also DO2/DO3 is controlled by the function block.	Boolean
Outputs (from top to bottom)	Rev Status (Close): If the motor is running in a reverse direction the output is set to logical 1.	Boolean
	Stop Status: If the motor has been stopped this output is set to logical 1.	Boolean
	For Status (Open): If the motor is running in a forward direction the output is set to logical 1.	Boolean
	Open: If the valve is open (i.e. end positions left) this output is set to logical 1.	Boolean
	Close: If the valve is closed (i.e. in end position) this output is set to logical 0	Boolean
	Rev Lockout Time Running: When logical 1 the lockout time is running and a start signal is ignored.	Boolean
	Out of Position: There is a positioning error.	Boolean
	Run Time Exceeded: Time between opening and closing is longer than permitted by the user configuration.	Boolean
	Status DO0 ... DO3: Mirror of the digital outputs of the UMC.	Boolean
Parameters	See section "Control Function Actuator 1-4" in the UMC100 manual.	

Function block reference

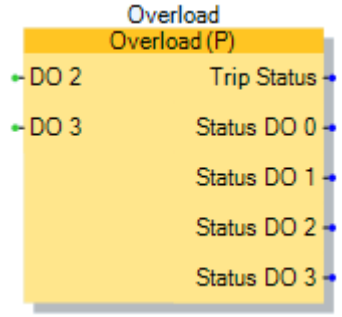
Transparent

Use this function if no control function shall be executed on the UMC but the outputs shall be controlled from a PLC.

Function Block		
	Signal Name	**Data Type**
Inputs (from top to bottom)	DO0 - DO3: If an input is logical 1 the corresponding output is switched on.	Boolean
Outtput (from top to bottom)	Status of the outputs.	Boolean
Parameters	See section "Control Function Transparent" in the UMC100 manual.	

Electronic Overload (EOL)

Use this function if a conventional TOL/EOL device shall be replaced with an UMC.

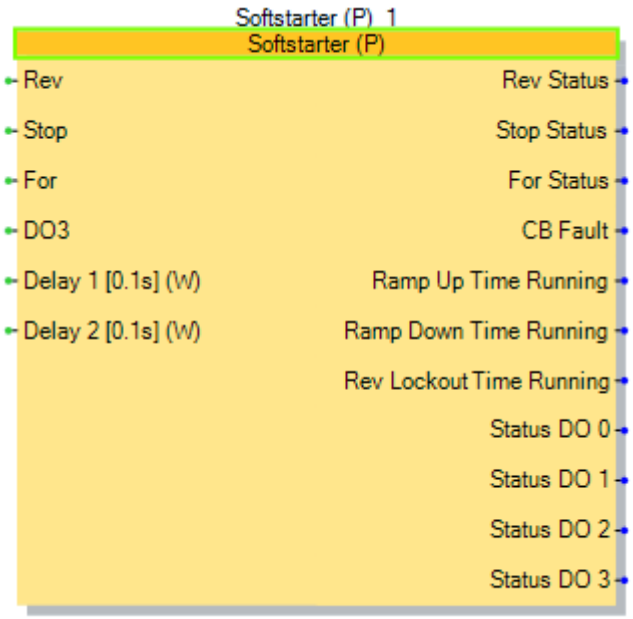
Function Block		
	Signal Name	**Data Type**
Inputs (from top to bottom)	DO2 - DO3: If an input is logical 1 the corresponding output is switched on.	Boolean
Outtput (from top to bottom)	Trip Status: Indication if a tip has happened (i.e. DO0 is opened and DO1 is closed)	Boolean
	Status of the outputs.	Boolean
Parameters	See section "Control Function Overload" in the UMC100 manual.	

Function block reference

Softstarter

Use this function to control a softstater in one or two directions of rotation. Usually this control function is used together with softstarters without own communication interface.

Function Block



	Signal Name	Data Type
Inputs (from top to bottom)	Rev: Run reverse signal, rising edge	Boolean
	Stop: Stop signal	Boolean
	For: Run forward signal, rising edge	Boolean
	DO3: Input to freely control DO3 if not used as fault output	Boolean
	Delay1: Delay for softstarter start output signal DO2 0...100 [0.1s], if input is >100 the value is set to 100	Boolean
	Delay2: Off-Delay for DO0 and DO1 after ramp down 0...100 [0.1s], if input is >100 the value is set to 100	Boolean
Outtput (from top to bottom)	Rev/Stop/For Status: Motor status	Boolean
	CB Fault: Checkback fault detected	Boolean
	Rev Lockout Time Running: Reversing lockout time is running	Boolean
	Ramp Up/Down Time Running: Ramping up, ramping down	Boolean
	Status DO0 ... DO3: Status of the outputs	Boolean
Parameters	See section "Control Function Softstarter" in the UMC100 manual.	

Function block reference

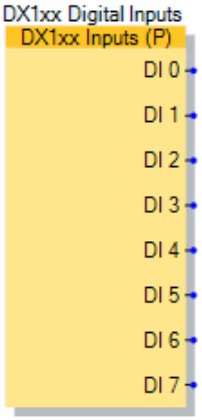
Function Blocks for Expansion Modules

The access to inputs, outputs and process data of the expansion modules is realised with function blocks. The following section describes these function blocks.

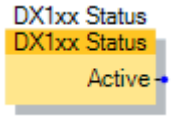
DX1xx Inputs

This function block provides access to the digital inputs of the modules DX122 and DX111

.

Function Block		
	Signal Name	**Data Type**
Outputs	DI0 ... DI7: Digital inputs on the module	Boolean
Inputs	-	
Parameters	DX1xx DI Delay	

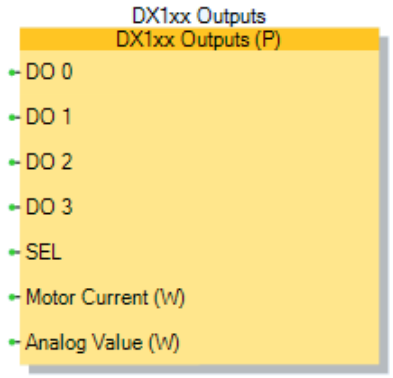
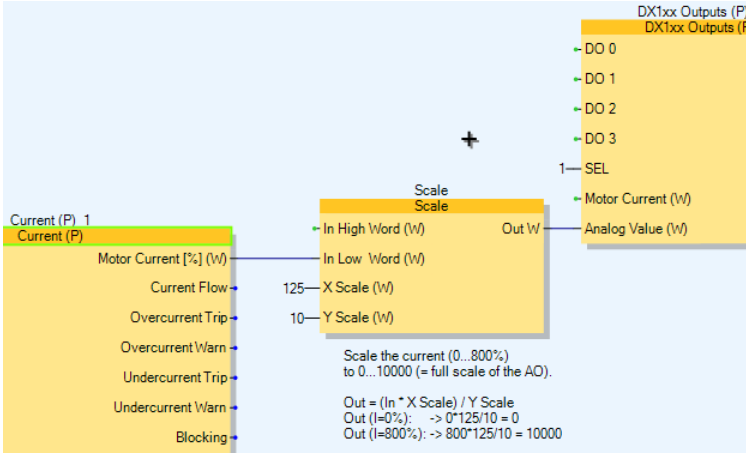
DX1xx Status

Function Block		
	Signal Name	**Data Type**
Outputs	Active: If true the DX1xx is connected and running.	Boolean
Inputs	-	-
Parameters	-	

Function block reference

DX1xx Output

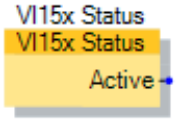
This function block provides access to the outputs of the digital I/O expansion module DX111 and DX122.

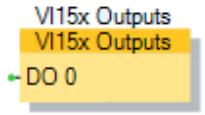
Function Block		
	Signal Name	**Data Type**
Outputs	-	-
Inputs	DO0 ... DO3: Relay outputs on the module	Boolean
	SEL: A logical 0 selects the 'Motor Current' input as source for AO+/AO-. A logical 1 selects the 'Analog Value' input as source for AO+/AO-. If the input is left open the 'Motor Current' is automatically selected.	Boolean
	Motor Current: Usually this input is connected to the 'Motor Current' output of function block 'Current'. The input value is automatically scaled in a way that an input value of 200 (i.e. 200% of I_e) results in 100% at the analog output (e.g. 20mA or 10V). Input values > 200 are limited to 200.	Word
	Analog Value: A value of 0 ... 10000 results in 0 ... 100% at the analog output (e.g. 4 ... 20mA). Use this output if you want to display an analog value on a pointer instrument or to send the process value to a PLC. Example: Display the motor current (0...800% of I_e) on a pointer instrument of type 0...10V. 	Word
Parameters	AnalogOut 0 Type	

Function block reference

VI15x

Function blocks for the modules VI150 and VI155

Function Block		
	Signal Name	**Data Type**
Outputs	Active: If true the VI15x is connected and running.	Boolean
Inputs	-	-
Parameters	-	

Function Block		
	Signal Name	**Data Type**
Outputs	-	-
Inputs	DO0: Set / reset the relay output on the VI15x module.	Boolean
Parameters	-	

Function block reference

VI15x Voltage

This function block provides access to all the process values and status signals from the voltage measurement module.

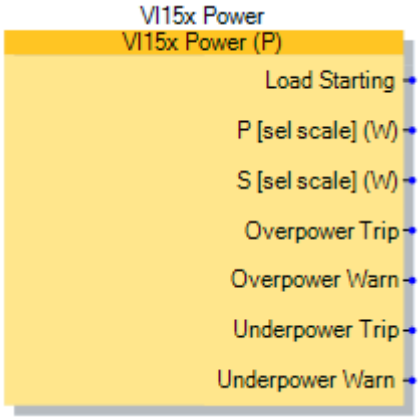
Function Block

	Signal Name	Data Type
Outputs	Average Voltage [V]: Average voltage of U_{L1L2}, U_{L2L3}, U_{L3L1}	Word
	UL1L2, UL2L3, UL3L1: Line voltages	Word
	Overvoltage Trip/Warn: True if one line voltage is above the adjusted overvoltage trip/warn level.	Boolean
	Undervoltage Trip/Warn: True if one line voltage is below the adjusted undervoltage trip/warn level.	Boolean
	Phase Loss: One or more phases are lost (e.g. fuse blown)	Boolean
	U Imb.: The voltage imbalance value	Word
	U Imb. Trip/Warn: True if the voltage imbalance is above the adjusted trip/warn level.	Boolean
	Wrong Phase Sequence: Output is set if a wrong phase sequence was detected.	Boolean
	THD L1, L2, L3: Total harmonic distortion on the phase.	Word
	THD Warn: Output is set if the total harmonic distortion is above the configured warn level in a phase.	Boolean
Inputs	-	-
Parameters	U Imb. Trip Level, U Imb. Trip Delay, U Imb. Warn Level, U Imb. Warn Delay, THD Warning Level, THD Warning Delay U Low Trip Level, U Low Trip Delay, U Low Warn Level, U Low Warn Delay, U High Trip Level, U High Trip Delay, U High Warn Level, U High Warn Delay	

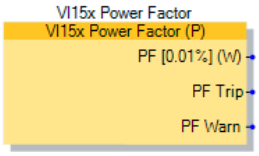
Function block reference

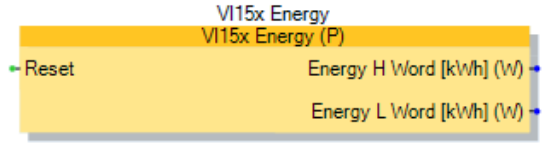
VI15x Power, PF and Energy

The following function blocks provide access to the power, power factor and energy process values and status signals.

Function Block		
	Signal Name	**Data Type**
Outputs	Load Starting: The load startup time is running	Boolean
	P [sel scale]: Internally the Active Power is represented as DWORD. To allow the transfer via fieldbus the internal value can be scaled with the parameter Power Scale Factor. This output provides the scaled active power value. See section "VX150 / VX155 Parameters" in the UMC100 manual for an example on how to use the scaling factor.	Word
	S [sel scale]: Apparent Power with the selected scaling factor applied.	Word
	Overpower Trip/Warn: The output is set if the active power is above the adjusted tip/warn level.	Boolean
	Underpower Trip/Warn: The output is set if the active power is below the adjusted tip/warn level.	
Inputs	-	-
Parameters	P Low Trip Level, P Low Trip Delay, P Low Warn Level, P Low Warn Delay, P High Trip Level, P High Trip Delay, P High Warn Level, P High Warn Delay Nominal Power Factor Power Scale Factor Load Startup Delay	

Function block reference

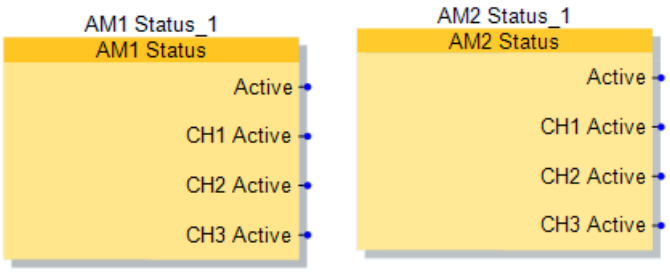
Function Block		
	Signal Name	**Data Type**
Outputs	PF: Power Factor (cos phi)	Word
	PF Trip: Output is true if the power factor is below the adjusted trip level.	Boolean
	PF Warn: Output is true if the power factor is below the adjusted warn level.	Boolean
Inputs	-	-
Parameters	PwrFactor Trip Level, PwrFactor Trip Delay, PwrFactor Warn Level, PwrFactor Warn Delay	

Function Block		
	Signal Name	**Data Type**
Outputs	Energy H Word: Internally the energy is represented as a DWORD. To transmit it via fieldbus it is split up into two words. This ouput represents the high word.	Word
	Energy L Word: Internally the energy is represented as a DWORD. To transmit it via fieldbus it is split up into two words. This is the low word.	Word
Inputs	Reset: If the input is set the energy counter is reset to zero. This allows to measure the energy consumed by a specific batch etc.	Boolean
Parameters	-	

Function block reference

AI111

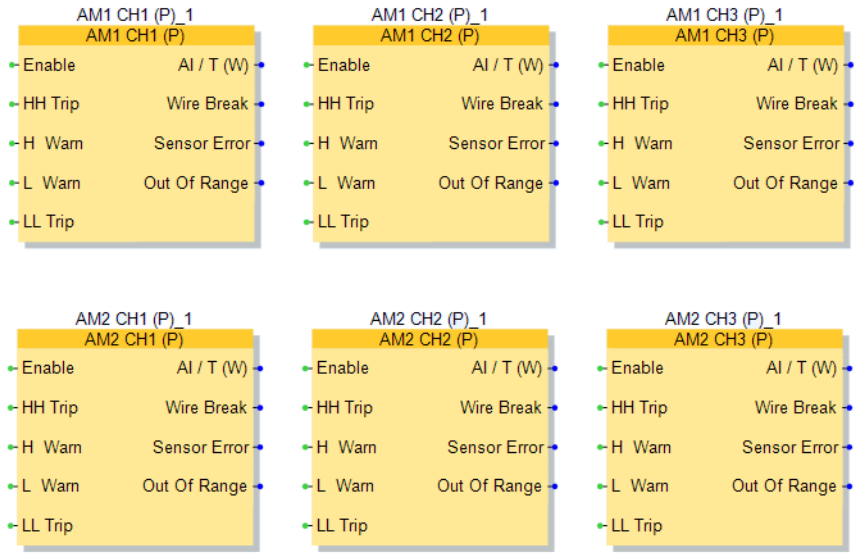
Function blocks for the analog input module AI111

Function Block		
	Signal Name	**Data Type**
Outputs	Active: True if the analog module 1/2 is configured and in data exchange	Boolean
	CH1 Active: True if channel 1 is s configured and the module is in data exchange	Boolean
	CH2 Active: True if channel 1 is s configured and the module is in data exchange	Boolean
	CH2 Active: True if channel 1 is s configured and the module is in data exchange	Boolean
Inputs	-	-
Parameters	-	

Function block reference

The functions blocks "AM1/2 Channel 1-3" should only be used in case of parameters "AM1 Mode" or "AM2 Mode" are set to "Universal". Otherwise the temperature function blocks "AM1/2 Temperature" shall be used.

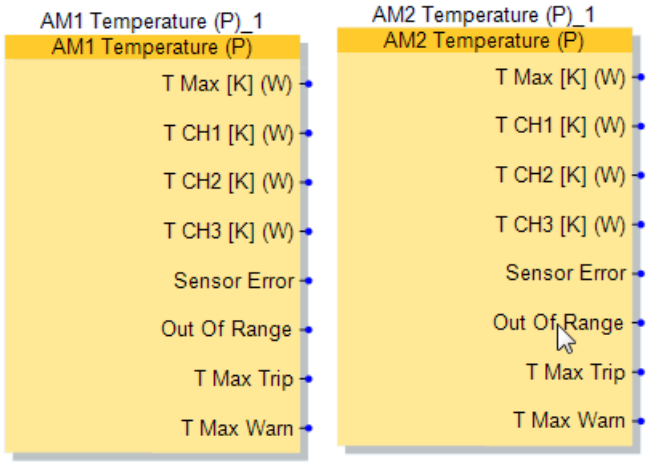
Function Block



	Signal Name	Data Type
Outputs	AI/T: Analoge input value channel 1, 2 or 3 (0…65535)	Word
	Wire break (only in configuration 4…20mA, value < 4mA)	Boolean
	Sensor Error: Temperature measurement is configured: short circuit or wire break detected during temperature measurement	Boolean
	Out Of Range: Value is out of range	Boolean
Inputs	DO0: Set / reset the relay output on the VI15x module.	Boolean
	Enable: Enable FB calculation. Allows to disable function block execution in certain operation states (e.g. startup ...)	Boolean
	HH/LL Trip: Triggers a channel related error message that the upper trip level / lower trip level was reached. Message is signalled on the display and fieldbus	Boolean
	H/L Warn: Triggers a channel related warning message that the upper warn level / lower warn level was reached. Message is signalled on the display and fieldbusus	Boolean
Parameters	AM1/2 Mode, AM1/2 Ch1/2/3 Error Reaction, AM1/2 CH1/2/3 Type	

Function block reference

The functions blocks "AM1/2 Temperature" should only be used in case of parameters "AM1 Mode" or "AM2 Mode" are set to "Temperature". Otherwise the AM1/2 Channel 1-3" function blocks shall be used.

Function Block	

	Signal Name	Data Type
Outputs	T Max [K] (W): Highest value of the 3 temperature input channels in kelvin	Word
	T CH1/2/3 [K] (W): Temperature of channel 1/2/3 in kelvin	Word
	Sensor Error: temperature measurement is configured; short circuit or wire break during temperature measurement detected	Boolean
	Out Of Range: value out of range	Boolean
	T Max Trip: Trip level of temperature reached	Boolean
	T Max Warn: Warn level of temperature reached	Boolean
Inputs	-	-
Parameters	AM1/2 Mode, AM1/2 Ch1/2/3 Error Reaction, AM1 Tmax Trip Level, AM2 Tmax Trip Level, AM1 Tmax Warn Level, AM2 Tmax Warn Level, AM1/2 CH1 Type	

Function block reference

Number of Functionblocks per Block Type

The following table shows how many blocks per type can be used in a customer specific application. If the count shows '-' the number of blocks is only limited by the available application memory.

Function Block Name	Max. number of blocks
Basic DI (P)	1
Binary Command 0	1
Binary Command 1	1
Binary Command 2	1
Binary Command 3	1
Analog Command 0	1
Binary Command (Acyclic) 0	1
Binary Command (Acyclic) 1	1
Analog Command (Acyclic) 2	1
LCD Panel Command	1
Binary Monitoring 0	1
Binary Monitoring 1	1
Binary Monitoring 2	1
Binary Monitoring 3	1
Binary Monitoring 4	1
Binary Monitoring 5	1
Analog Monitoring 0	1
Binary Monitoring (Acyclic) 0	1
Binary Monitoring (Acyclic) 1	1
Analog Monitoring (Acyclic) 2	1
LCD Panel Monitoring	1
DX1xx Inputs (P)	1
DX1xx Status	1
DX1xx Outputs (P)	1
VI15x Status	1
VI15x Voltage (P)	1
VI15x Power Factor (P)	1
VI15x Power (P)	1

Function Block Name	Max. number of blocks
VI15x Energy (P)	1
VI15x Outputs	1
OR 2	-
OR 3	-
OR 4	-
AND 2	-
AND 3	-
AND 4	-
XOR 2	-
XOR 3	-
XOR 4	-
RTrig	5
FTrig	5
RS	10
CTUD	4
TON	10
TOF	10
Flash	4
Limit	6
Max	5
Min	5
Scale	5
SR	10
Non Volatile SR	4
Aux Fault Warn (P)	1
General Fault Warn	1
Thermal Load (P)	1
Thermistor (P)	1

Function Block Name	Max. number of blocks
Earth Fault (P)	1
Current (P)	1
Number Of Starts (P)	1
Prepare Emergency Start (P)	1
Fault Reset	1
Param To Bit (P)	1
Single Fault Reset	1
Physical Block (P)	1
Absolute Current (P)	1
Command Control (P)	1
Transparent (P)	1
Overload (P)	1
Direct Starter (P)	1
Rev Starter (P)	1
Star Delta Starter (P)	1
Pole Changing Starter (P)	1
Actuator 1 (P)	1
Actuator 2 (P)	1
Actuator 3 (P)	1
Actuator 4 (P)	1
Analog Mux	4
Scale1	6
ParamToPV1-5	1 each
SelectPV	1
AM1 Status	1
AM2 Status	1
AMx Channelx	1 each
AMx Temperature	1 each

Info and error messages

During editing and on user request the custom application is checked against a number of design rules to ensure a valid application. Design rules can either lead to an error message or a warning.
Messages are displayed in the message window.
Messages are not stored and are deleted if the configuration application is closed.
Each warning and error is uniquely identified with a number.
The following table lists all messages with possible corrective measures.

Error / Warning Code	Explanation / Corrective Measure
1000	Connection of a boolean port to a word port. Only connections between ports of the same data types are allowed.
1001	The size of an application is limited. The limit was reached.
1002	A function block has a port which is no longer available in the presently selected function block library. Select the function block library version that was used when the diagram was created. Or delete the function block and replace it with the version available in the presently selected library.
1003	The source block of a connection is missing.
1004	The sink block of a connection is missing.
1005	The port cannot be negated.
1006	An application requires exactly one function block from group "Control Functions".
1007	A function block was not added to the program because either all the inputs or outputs are disconnected. i.e. the function block is meaningless.
1008	A constant value is not supported at this position.
1009	The application running in the UMC is different from the one in the application editor (checksum differs).
1010	Time constants are internally handled with a resolution of 100ms. The provided value was inbetween two 100ms steps and was rounded to the next 100ms step.
1011	The given delay time was larger than the max. supported value.
2001	The number of instances per function block type is limited. The limit for a certain type of function block was reached.
2002	The size of an application is limited. The limit was reached.
2003	An application requires exactly one function block from group „Control Functions“.
2004	A design rule was violated (e.g. number of function blocks).
2005	The instance name of each function block must be unique. By default unique names are choosen. If they are modified by hand unique names must be used.
2006	The selected starter function block is different from the selected control function. Both must match.
3001	Internal error. Please report it to your responsible sales contact. Use the template provided at the end of the document.
3002	There is an inconsistency between the application and the selected function block version. Ensure that the library version is selected that was used when the application was created.
3003	No function block library version was selected. Select the version supported by the used UMC.
3004	A previously exported application could not be imported anymore. Most probably the file is corrupt.
3005	A previously exported application could not be imported anymore. Most probably the file is corrupt.
3006	There is an inconsistency between the application and the selected function block version. Ensure that the library version is selected that was used when the application was created.
9999	Internal error. Please report it to your responsible sales contact. Use the template provided at the end of the document

Info and error messages

Application Run-Time Error Codes

In the event of a Custom Application Error which is shown on the UMC panel a detailed error code can be found in the FIM diagnostic mode on tab card diagnostics 2 (the two values on the bottom of the window). The following table explains theses values:

Custom Application Status Byte 0, Bit Number	**Explanation / Corrective Measure**
0	Unknown function block. This might happen if a UMC received a function block application which was created for an incompatible device version. Ensure that the firmware and function block library fit together. Re-check your application and ensure that only supported function blocks are used.
1	Application code too large. Ensure that the firmware and function block library fit together.
2	Memory full. Decrease the number of function blocks that require instance data (e.g. counters, timers)
3-6	Reserved
7	Internal error
Custom Application Status Byte 1, Bit Number	
1	Checksum error. Please reload the application into the UMC.
2	Wrong UMC version. The application was compiled for a different UMC version. Ensure that the firmware and function block library fit together.
3	Wrong runtime system version. The application was compiled for a different UMC version. Ensure that the firmware and function block library fit together.
4	Wrong device description file version. The application was compiled for a different UMC version. Ensure that the firmware and function block library fit together.
5	Timeout during application timeout. Reload the application.
6	Application too large. As the Custom Application Editor checks the max. possible size the application was compiled for a different UMC version. Ensure that the firmware and DTM version fit together.
7	Motor control function block not as paramtriesed in the parameters. The configured motor control function must fit to the used control function block.

Detected an error?

FAX No.: +49 (0) 6221-701-1382

Your feedback helps us to constantly improve our products. We are grateful for your comments and suggestions. Please provide us with the following information if you have noticed an issue:

Name

Company / Department

Telephone / Email

Problem Description

- Steps to reproduce the problem
- Version of UMC (Ident number on nameplate and firmware version which is displayed on the UMC100-PAN)
- Version of FIM and UMC100.3 Device Package

—
ABB STOTZ-KONTAKT GmbH
Electrification Products Division
Low Voltage Products and Systems
Eppelheimer Strasse 82
69123 Heidelberg, Germany

You can find the address of your local sales organization on the ABB homepage

abb.com/lowvoltage

Additional information
We reserve the right to make technical changes or modify the contents of this document without prior notice. With regard to purchase orders, the agreed particulars shall prevail. ABB AG does not accept any responsibility whatsoever for potential errors or possible lack of information in this document.

Document number 2CDC135034D0201 Rev.B (03/2019)