

Altivar 32

Function Blocks

02/2010



Draft
2010_02_01
for internal use only

Informations :

- Sur la version du logiciel reçu, les essais en réel sur un ATV32 n'étaient pas réalisables, ainsi cette aide en ligne reprend les infos connues à ce jour.

- La mise en page n'est pas définitive. La nouvelle charte sera appliquée lors de la saisie PAO.

-Pour les informations surlignées en jaunes nous attendons les précisions ou confirmations de la part du projet

Schneider
Electric

Overview of Function Blocks	5
Creating a FBD application in the Function Blocks	7
Insertion of function blocks	7
Creation of Input or Output	7
Links between Function Blocks	8
Function Blocks configuration.....	9
Function Blocks configuration.....	9
Configuration: Input or Output	10
Type of Link	11
Display Options	11
Palette panel	11
Menu bar	12
Import / Export:	13
Edition.....	13
Find:.....	15
Set / View execution order:	16
Reset execution order:	16
Check the application:	17
Manipulating FBD objects	18
Simulation:	19
Monitoring:	19
Mode:.....	19
ATV:.....	19
Zoom:	19
Configure:.....	21
Function Blocks Password:	22
Wiring sheet dimensions.....	23
FBD language elements	24
Special Inputs Function.....	24



NUM (Signed) 24



NUM (Unsigned) 24



TRUE 25



FALSE 25







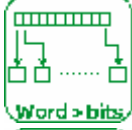
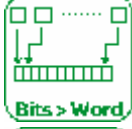





Logic Function..... 26



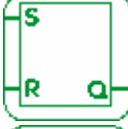





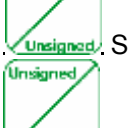
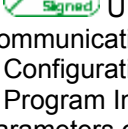


The NO function, 26



The AND function, 26

	The OR function,	26
	The NO AND function,	27
	The NO OR function,	27
	The EXCLUSIVE OR function.	27
	ADD/SUB (ADD/SUB Arithmetic Function).....	30
	Boolean Function	31
	CAN (Word to Bit Conversion).....	32
	CNA (Bit to Word Conversion).....	32
	COMPARE (Comparison of two values).....	33
	Edge.....	34
	GAIN Function	35
	MUL/DIV (MUL/DIV Arithmetic Function)	36
	MUX (Multiplexing).....	37

	ReadBitParam.....	38
	ReadParam.....	39
	RS memory.....	40
	Shift/Roll.....	41
	TRIGGER (Schmitt Trigger).....	42
	UP/DOWN and PRESET COUNT.....	44
	WriteBitParam.....	47
	WriteParam.....	48
	STU (Signed 16bits TO Unsigned 16Bits).....	49
	UTS (Unsigned 16bits TO Signed 16Bits).....	50
	Communication option.....	51
	Configuration of connection.....	51
	Program Import.....	51
	Parameters of communication.....	52
	Description of ADL Containers.....	52
	FB SYSTEM WORDS.....	53
	Supervision/Monitoring Window.....	54

Overview of Function Blocks

The Logic function provides a graphic programming based on predefined function blocks. The graphic programming is FBD language (Function Block Diagram), and there is a range function blocks: timer, counter, logic, etc.

The program in the edit window corresponds to:

- Compiled program,
- Program transferred in the ATV32.

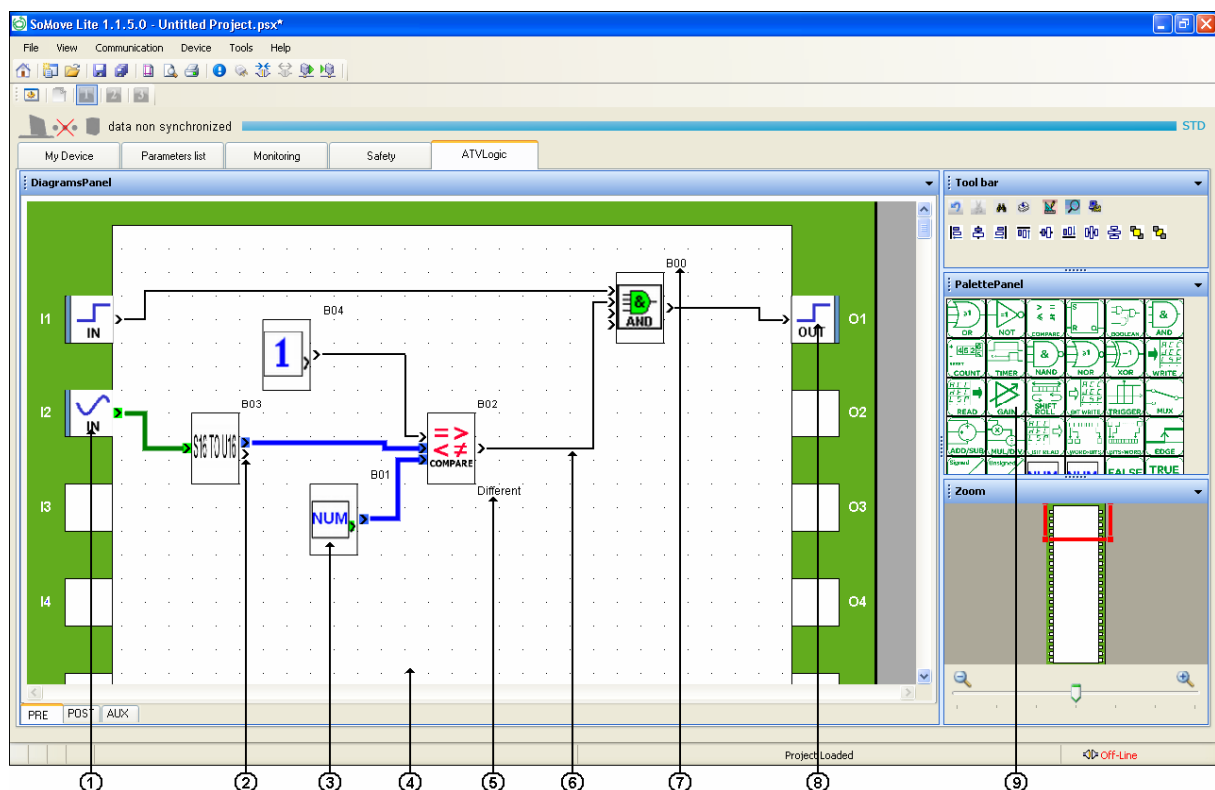
To create an FBD (Function Block Diagram) program, insert various function blocks in the wiring sheet, and link them together.

All types of function blocks can be placed on the sheet, except for the inputs and the outputs.

The edit window is made up of three zones:

- The wiring sheet, where the functions that make up the program are inserted,
- The Inputs zone on the left of the wiring sheet where the inputs are positioned,
- The outputs zone on the right of the wiring sheet where the outputs are positioned.

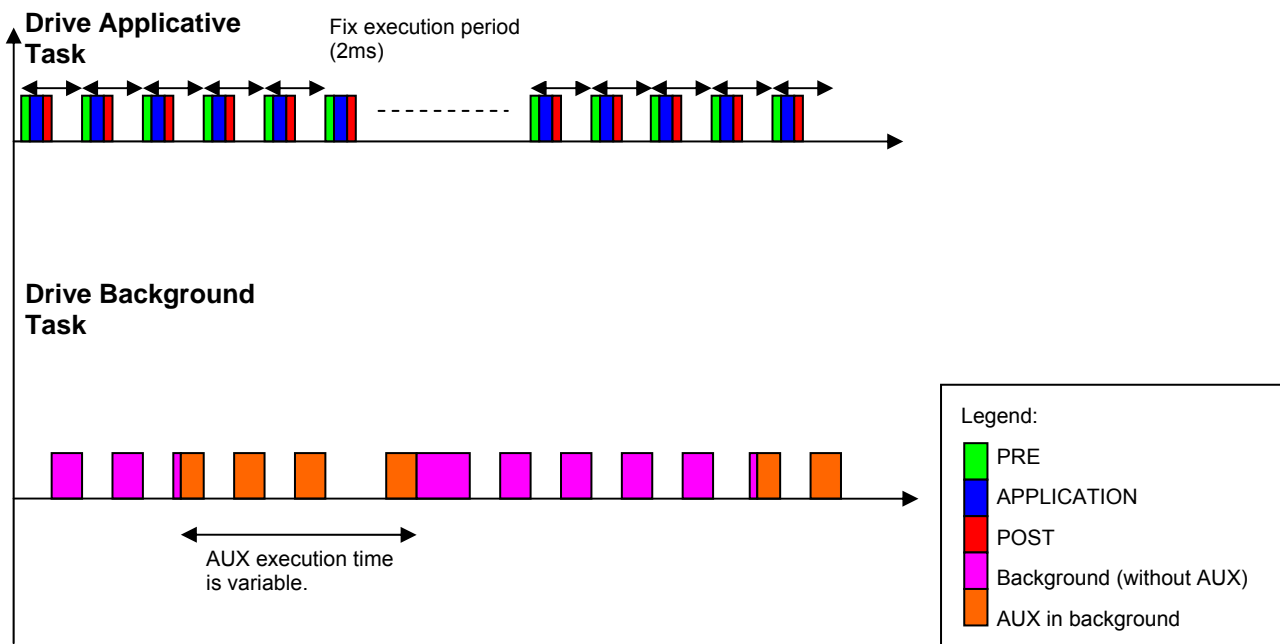
The following figure shows an example of a part of an edit window in FBD language:



- | | |
|-------------------------------|---------------------------------------|
| 1) Function block input zone. | 6) Links between two function blocks. |
| 2) Connection. | 7) Function blocks number. |
| 3) Function block. | 8) Function block Output zone. |
| 4) Wiring sheet. | 9) Palette pan |
| 5) Comments. | |

The Function Blocks provide a graphic programming based on predefined function blocks. In Function Blocks, there are three types of task:

Pre	The PRE task is executed in priority because it will reformat the inputs and motor control. The cycle of the task is set to 2ms.
Post	The Post task is executed in priority because it resort information related to motor operation. The cycle of the task is set to 2ms.
Aux	The AUX task runs in the background task because it deals with the heavy actions. It can be slowed by using communication.



The Pre and Post task are executed on average each 500µs and can not exceed 2ms. This leaves an average of 1ms for perform the task Aux. The task is executed the maximum of time, during the time allotted. But if the task exceeds 1ms it will perform several times.

Creating a FBD application in the Function Blocks

Insertion of function blocks

To add a function blocks, drag the function blocks that you wish to add on the workspace.

Creation of Input or Output

An input or output is a virtual port. It can be moved along the side of the wiring sheet.



- 1) Input port
- 2) Output port

There are 2 types of input / output virtual analog and the register.

The type of the block can be configured by double clicking on it.

[Logic input 1 assignment] (IL01) to [IL10] (IL10): function blocks logical input,
[Analog input 1 assignment] (IA01) to [IA10] (IA10): function blocks analog input,
[Logic output 1 assignment] (OL01) to [OL10] (OL10): function blocks logical outputs,
[Analog output 1 assignment] (OA01) to [OA10] (OA10): function blocks analog outputs.

When you double click on a virtual port, only this port can be changed, the others are grayed out.

For example, configuration of the input 3:

The screenshot shows the 'Virtual Port Configuration' dialog box. It has a table with columns 'Port', 'Data Type', and 'Virtual Number'. The 'Data Type' dropdown for port I3 is open, showing options: 'Discrete', 'Analogic', and 'Not implemented'. A box labeled '1' points to the 'Data Type' dropdown, and a box labeled '2' points to the 'Virtual Number' column.

Port	Data Type	Virtual Number
I1	Not implemented	1
I2	Discrete	0
I3	Analogic	0
I3	Not implemented	0
I4	Not implemented	0
I5	Not implemented	0
I6	Not implemented	0
I7	Not implemented	0
I8	Not implemented	0
I9	Not implemented	0
I10	Not implemented	0

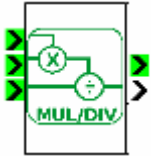
- 1) Choice of the data type,
- 2) Number of the virtual output.

Links between Function Blocks

Position the function blocks in the wiring sheet, then link them together. Link one block output to an input of another block or loop an output back to an input of the same block.

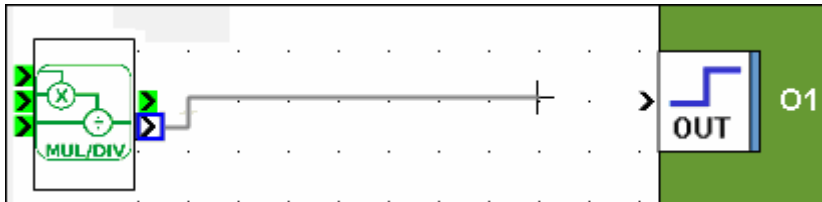
The following procedure describes how to link function blocks together:

1. Left click on an output of a function block.
Result: the mouse cursor is displayed as a cross.





Keep the left button held down.

2. With the button held down, move the cursor over a block input.
Result: the mouse cursor is displayed as a cross.



3. If the cursor turns into a circle with a line through it when you move it over a block input, this means that the link destination is incorrect (incompatible types).

- If the connection is available, the cursor is 
- If the connection is forbidden, the cursor is 

Release the mouse button.

Result: The line is shown between the two linked blocks.

Repeat steps 1 to 3 to link all the blocks.

Function Blocks configuration

Function Blocks configuration

Each of the function blocks has a configuration window. This window consists of one or two tabs:

- Comments for all function blocks,
- Block number,
- Parameters depending on the function blocks type (FBD PRESET COUNT).

When you double click on a function block, a configuration windows appears.

1) Comments tab:

All of the function blocks can have an associated comment. These comments are displayed under the block in the wiring sheet.

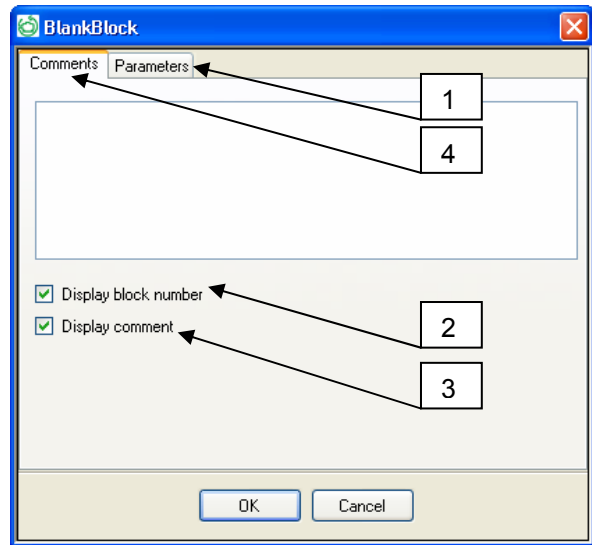
In the Comment zone you can enter a comment of up to **three lines of a maximum of 90 characters maximum.**

2) Display block number:

Display the block number permanently.
This option is activated by default.

3) Display comment:

When a comment has been added to a function block, If the "Display comment" box is checked the comment is permanently displayed.



4) Parameters tab:

Most function blocks have a parameters tab. In this tab, you have to set the function block's specific parameters. These parameters are described in detail in the help for each of the blocks.

Configuration: Input or Output

Each Input or Output has a configuration window.

With a double click on the departure or arrival  a configuration window appears.

Comments tab:

These comments are displayed under the block in the wiring sheet.

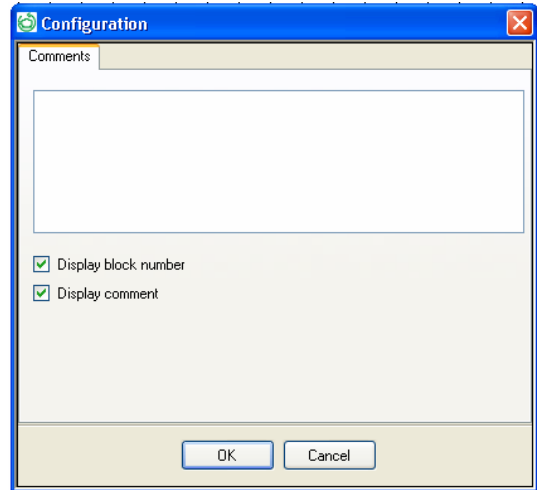
In the Comments zone you can enter a comment of up to **three lines of a maximum of 90 characters maximum.**

Display block number:

Display the block number permanently.
This option is activated by default.

Display comment:

When a comment has been added to a function block,
If the “Display comment” box is checked the comment is permanently displayed.



Type of Link

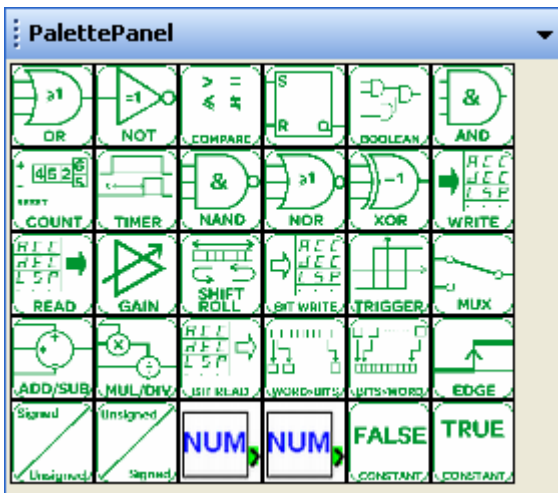
The link is displayed in different ways depending on the type of data transferred

- For a Boolean link it is a Black line (Boolean signal): _____
 - Possible value: 1 or 0.
- For a S16 link it is a Green line (analog signal): _____
 - Possible value: -32768 to 32767.
- For a U16 link it is a Blue line (register signal): _____
 - Possible value: 0 to 65535.

Display Options

Palette panel

To create an FBD program, the different possible functions are available in the palette panel. Each picture in the **palette panel** represents a function type.

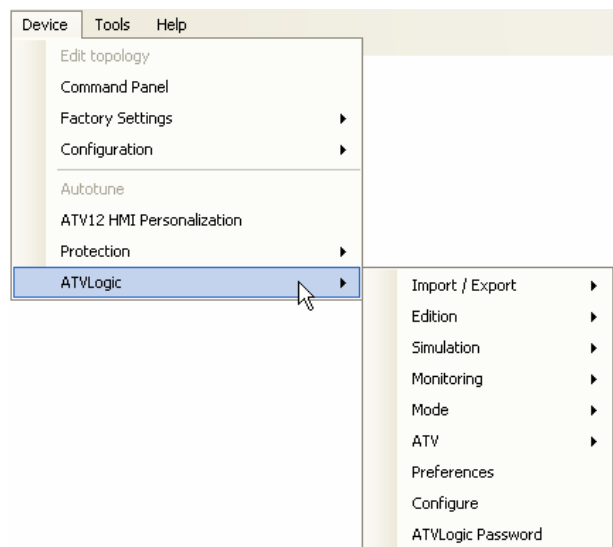


When the mouse is moved over one of the function type, the name of the function is view.

Menu bar

For Logic functions, several different options are available:

- Import / Export
 - Export
 - Import
- Edition
 - Cut
 - Cancel
 - Select all
 - Find
 - Set / View execution order
 - Reset execution order
 - Check the application
 - Align left
 - Align center
 - Align right
 - Align top
 - Align middle
 - Align bottom
 - Horiz. spacing
 - Vert. spacing
 - Bring to front
 - Send to back
- Simulation
 - InitRun
 - Pause
 - Stop
 - PowerOFF
 - ATVView
- Monitoring
 - Monitor
- Mode
 - Edit
 - Monitor
 - Simulate
- ATV
 - ATVcontrol
 - ComparePCtoATV
 - DeleteATVapplication
- Zoom
- Preferences
- Configure
- Password



Import / Export:

Export

Waiting information

Import

Waiting information

Edition

Cut:

This feature is accessible via the menu bar: **Function Blocks** → **Edition** → **Cut**, it allows you to cut one or more selected objects, it is too accessible with the  icon.

Cancel

This feature is accessible via the menu bar: **Function Blocks** → **Edition** → **Cancel**, it allows you to cancel the last action perform. It is too accessible with  icon.

How to select, deselect and delete one or more blocks?

How select one isolated block:

By clicking on isolated block function with the left-click of the mouse. Extend the selection to another block, the old block will be deselected.

To add an element to the selection, hold the key "SHIFT" and click on an element.

How select more objects:

To select several contiguous objects, Frame the objects to be selected by defining a selection zone.

To select several objects scattered about in the wiring sheet, press the Shift key, then click on the objects to be selected while continuing to hold down the Shift key.

How deselect one or more block:

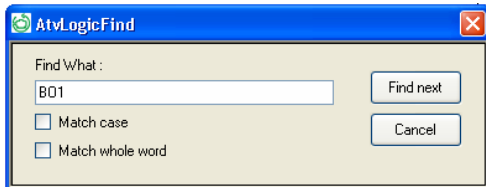
By clicking on a free area in the wiring sheet or press the Escape key, all the blocks will be deselected.

To deselected one block of the selection, press the Ctrl key and click on the objects that you want to deselect.

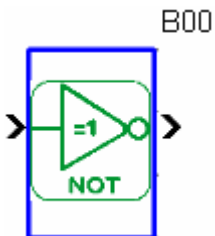
Find:

The find feature allows searching in: comments, function blocks, block number.

This function is accessible via the menu bar: **Function Blocks** → **Edition** → **Find**.



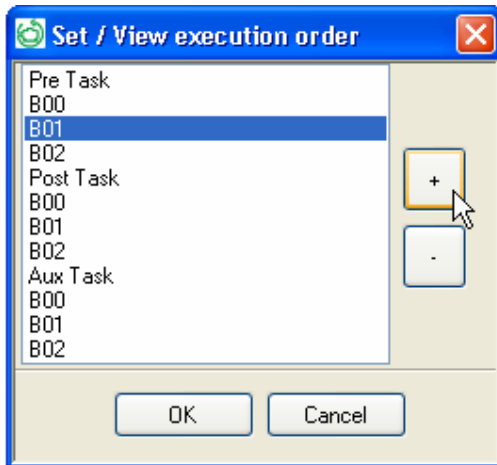
Once the research started, when a block meets the search criteria, it will be marked in blue. To see the other blocks that could meet the criteria, click again to find next.



Set / View execution order:

This feature sets the order execution of the function blocks.

This function is accessible via the menu bar: **Function Blocks** → **Edition** → **Set / View execution order**.




To restore the default values see "Reset execution order".

Reset execution order:

This feature resets the order execution of the function blocks.

This function is accessible via the menu bar: **Function Blocks** → **Edition** → **Reset execution order**.

Check the application:

With the **Edit** → **Compilation** command or , you launch the compilation (check) of the program. The result of the compilation is displayed in the compilation report window. In this window, the following information appears:

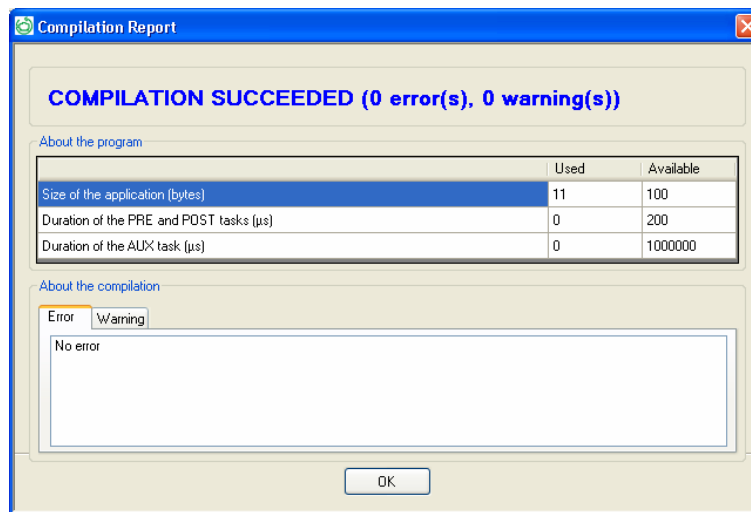
- The result of the program check,
- Resources used and available.

Results Window Elements

The compiler calculates the volumes of resources used in the drive.

If the values calculated are greater than the available values, they appear in the compilation report window.

The compilation report window displays the various possible errors recovered by compiling



List of errors:

Test	Error type	Consequences
The destination file can't be opened	Error	stop the compilation
The internal parameter has no value	Warning	the value is set to 0 by default
At least one virtual input block is present but no input block is connected	Error	/
At least one virtual output block is present but no output block is connected	Error	/
2 FB have the same execution order	Error	Normally prevented by HMI
2 inputs or outputs block have the same ID	Warning	/
1 input of a FB is not connected	Warning	the value is set to 0xFF
1 output of a FB is not connected	Warning	the value is set to 0xFF
Memory overflow	Error	/
Pre + Post duration	Error	/
Aux duration	Error	/
Binary file too large	Error	/
RAM allocation exceed	Error	Not treated yet
ADL container used more than 1 time	Warning	Not treated yet
FB that have inputs but no one connected	Error	/







Manipulating FBD objects

How to Align a Group of Objects

Select the objects to align.

Result: All of the selected objects are highlighted by small yellow squares placed at each corner of the block.

From the Align command in the Tools menu, select:



Icon	Title	Description
	Align left	Align the group of block vertically to the left.
	Align center	Align the group of block vertically to the center.
	Align right	Align the group of block vertically to the right.
	Align top	Align the group of block horizontally to the top.
	Align middle	Align the group of block horizontally to the middle.
	Align bottom	Align the group of block horizontally to the bottom.

How to Center a Group of Objects

Select the objects to center.

Result: All of the selected objects are highlighted by small yellow squares placed at each corner of the block.

From the center command in the Tools menu, select:



Icon	Title	Description
	Horizontal spacing	Center the group of block horizontally, equally spaced.
	Vertical spacing	Center the group of block vertically equally spaced.

How to Bring or Send an object to the Foreground or Background

Select the objects to bring or send.

Result: One or all of the selected objects are highlighted by small yellow squares placed at each corner of the block.

From bring or send command in the Tools menu, select:

Icon	Title	Description
	BringToFront	Bring an or the group of block to the foreground.
	SendToBack	Send an or the group of block to the background.

Simulation:

InitRun
Pause
Stop
PowerOFF
ATView

Monitoring:

Monitor

Mode:

Edit
Monitor
Simulate

ATV:

ATVcontrol
ComparePCtoATV
DeleteATVapplication

Zoom:

This function is accessible via the menu bar: **Function Blocks** → **Edition** → **Zoom**.

It gives you the choice whether to display the zoom window.

Preferences:

This function is accessible via the menu bar: **Function Blocks → Preferences.**

This feature edits:

- Workspace,
- Functions,
- Check report.

Workspace:

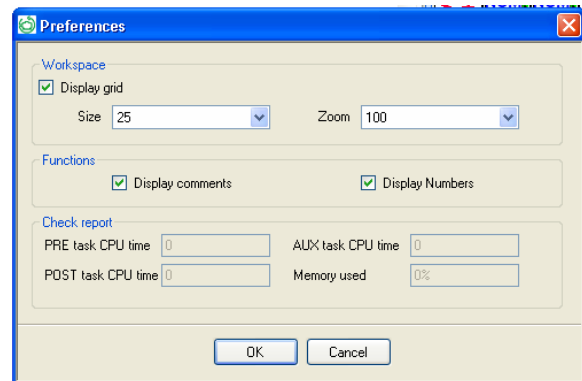
- Display or not the grid
- Choose the size of spaces
- Choose the zoom.

Functions:

All function blocks can have an associated comment and block number. They are shown above and below the function block in the wiring sheet.

You can choose to display:

- All comments,
- All numbers.



Configure:

This function is accessible via the menu bar: **Function Blocks → Configure.**

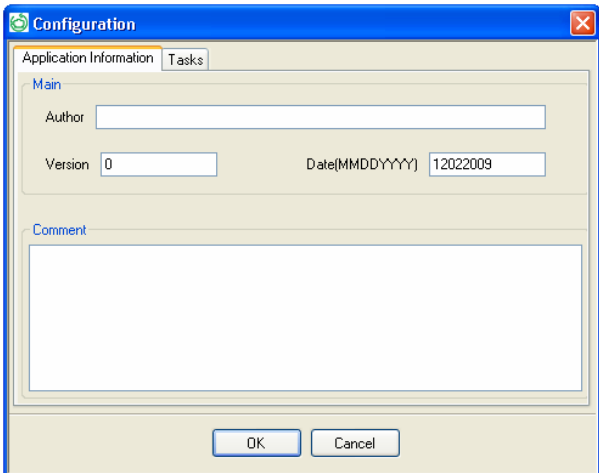
This window has two tabs:

- Application information,
- Tasks.

Application information:

This tab lets you enter information related to the program:

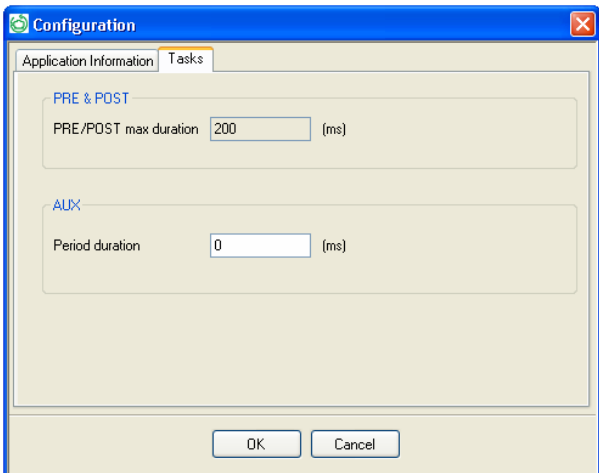
- The name of the Programmer,
- The version,
- The date,
- And a comment.



The screenshot shows the 'Configuration' dialog box with the 'Application Information' tab selected. The 'Main' section contains three input fields: 'Author' (empty), 'Version' (0), and 'Date(MMDDYYYY)' (12022009). Below this is a 'Comment' section with a large empty text area. At the bottom are 'OK' and 'Cancel' buttons.

Tasks:

This tab specifies the maximum time authorized for the conduct of the stain AUX. This can be likened to a WATCHDOG action (module cycle time control).

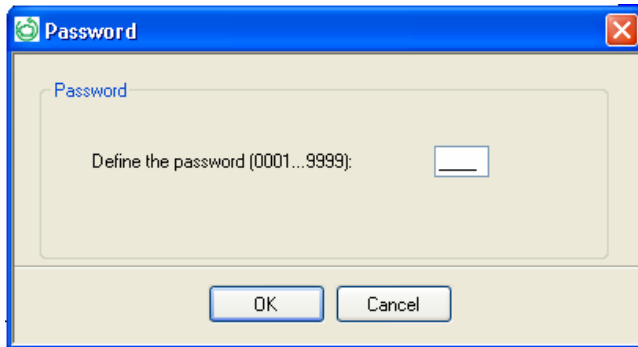


The screenshot shows the 'Configuration' dialog box with the 'Tasks' tab selected. The 'PRE & POST' section has a 'PRE/POST max duration' field set to 200 (ms). The 'AUX' section has a 'Period duration' field set to 0 (ms). At the bottom are 'OK' and 'Cancel' buttons.

Function Blocks Password:

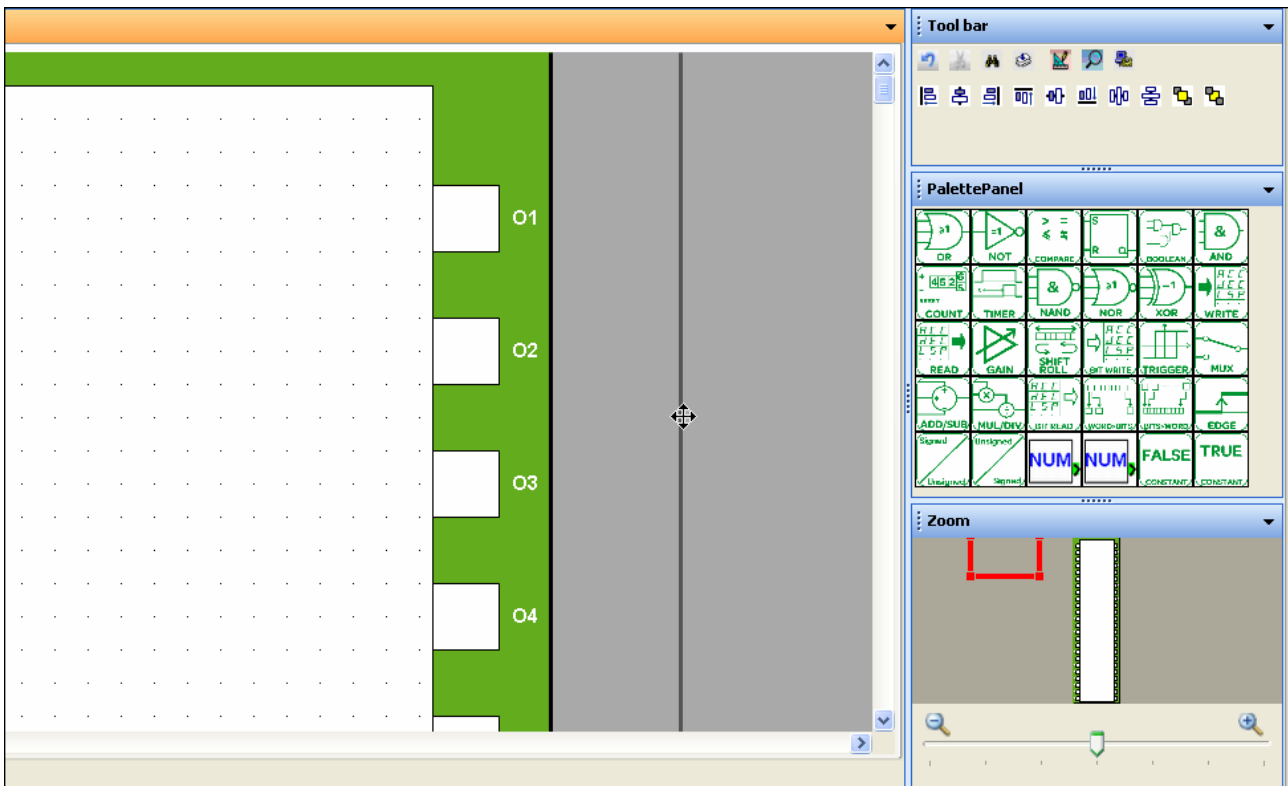
This function is accessible via the menu bar: **Function Blocks** → **Function Blocks Password**.

Define the password, which prevents the reading of your program.



Wiring sheet dimensions

It is possible to change the dimensions of the wiring sheet.
Resize the wiring sheet by clicking on the right side of it and drag it to the left or right.



FBD language elements

Special Inputs Function



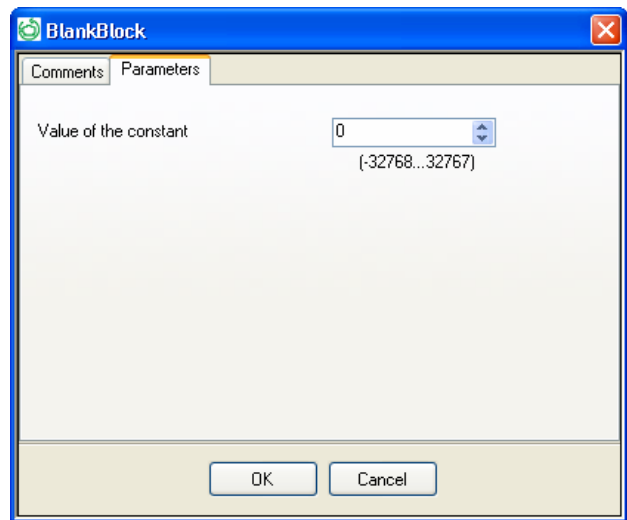
NUM (Signed)

The numerical constant NUM is an integer signed with a value between -32768 and +32767. This constant can be used to set values to the functions' non-connected inputs:

- GAIN,
- COMP IN ZONE,
- TRIGGER.

The value of the constant can be set in the Parameters window.

In Simulation or Monitoring modes, it is possible to modify the constant.

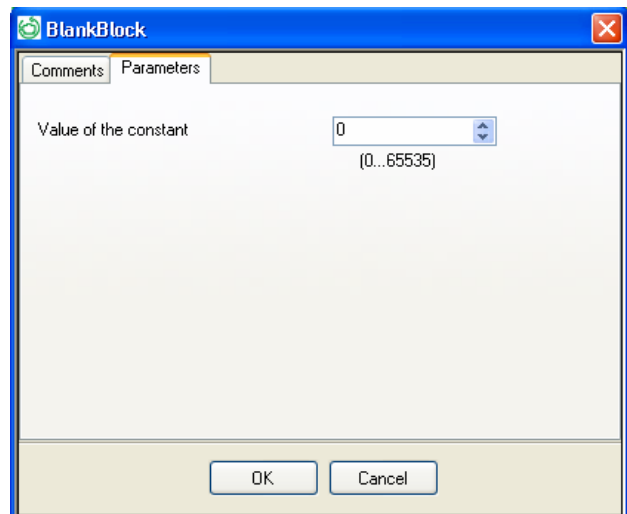


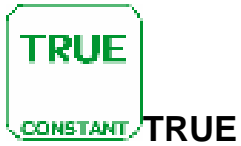
NUM (Unsigned)

The numerical constant NUM is an integer unsigned with a value between 0 and 65535.

The value of the constant can be set in the Parameters window.

In Simulation or Monitoring modes, it is possible to modify the constant.





This constant can be used to set the function inputs to 1.

In Simulation or Monitoring modes, you can force these inputs in the reverse order.



This constant can be used to set the function inputs to 1.

In Simulation or Monitoring modes, you can force these inputs in the reverse order.

Logic Function

In FBD language, it is possible to use logic functions in the block diagrams. These functions are accessible from the **Palette panel**. The available functions are:



The NO function,

If the input is inactive or not connected, the output is active.
If the input is active, the output is inactive.

- [IN]: A Boolean input
- [Q]: A Boolean output

In	Q
0	1
1	0



The AND function,

If all the inputs are active or not connected, the output is active.
If at least one input is inactive, the output is inactive.

- [IN1] to [IN4]: 4 Boolean input
- [Q]: A Boolean output.

In1	In2	In3	In4	Q
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1



The OR function,

If at least one input is active, the output is active.
If all the inputs are inactive or not connected, the output is inactive.

- [IN1] to [IN4]: 4 Boolean input
- [Q]: A Boolean output.

In1	In2	In3	In4	Q
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1



The NO AND function,

If at least one input is inactive, the output is active.
 If all the inputs are active or not connected, the output is inactive.

- [IN1] to [IN4]: 4 Boolean input
- [Q]: A Boolean output.

In1	In2	In3	In4	Q
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0



The NO OR function,

If all the inputs are inactive or not connected, the output is active.
 If at least one input is active, the output is inactive.

- [IN1] to [IN4]: 4 Boolean input
- [Q]: Boolean output.

In1	In2	In3	In4	Q
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0



The EXCLUSIVE OR function.

If an input is inactive and the other input is active or not connected, the output is active.
 If both inputs are active or inactive or not connected, the output is inactive.

- [IN1] and [IN2]: 2 Boolean input
- [Q]: Boolean output.

In1	In2	Q
0	0	0
0	1	1
1	0	1
1	1	0



TIMER A/C (Timer)

The Timer function is used to delay, prolong and control actions over a predetermined time. The Timer has three functions:

- The A function: timer on-delay, or timer active,
- The C function: timer off-delay, or timer idle,
- The A/C function: combination of functions A and C.

Inputs/Outputs:

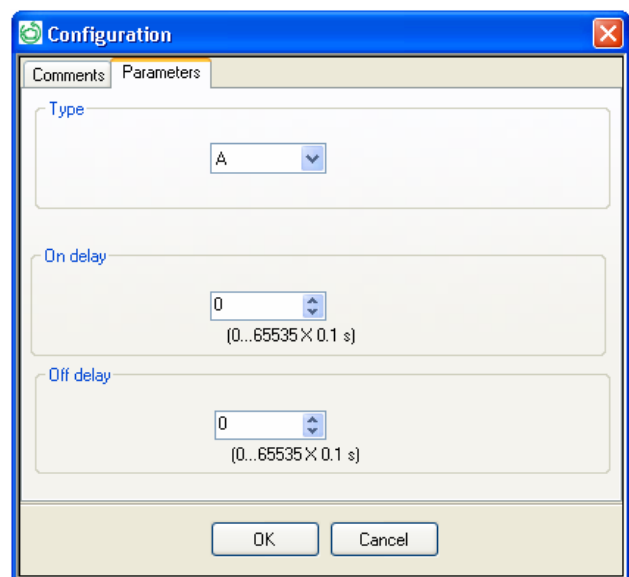
- [IN]: A Boolean input,
- [Q]: A Boolean output.

Configuration:

From the Configuration window, you can adjust in Parameters tab:

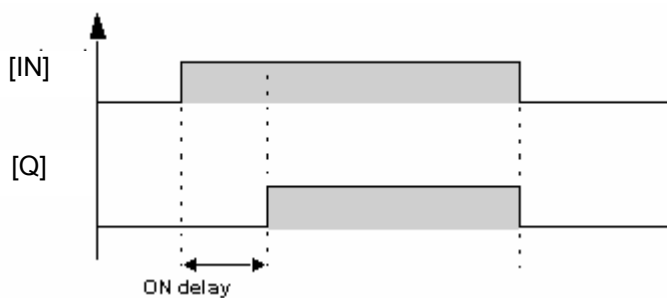
In the Parameters tab, you can adjust the value of the delays for each of the functions (A, C and A/C).

- ON delay for function A,
- OFF delay for function C,
- A combination of both the ON and OFF delays can be used to adjust function A/C.



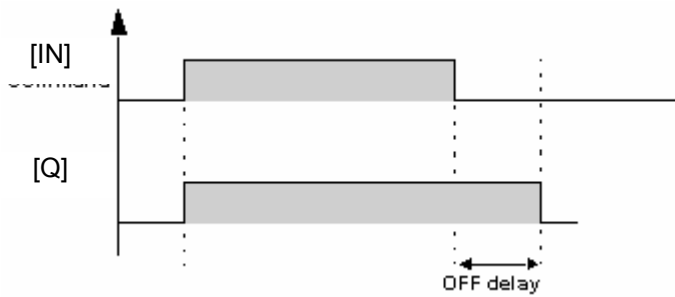
A Function

The following diagram shows the operation of the timer in function A:



C Function

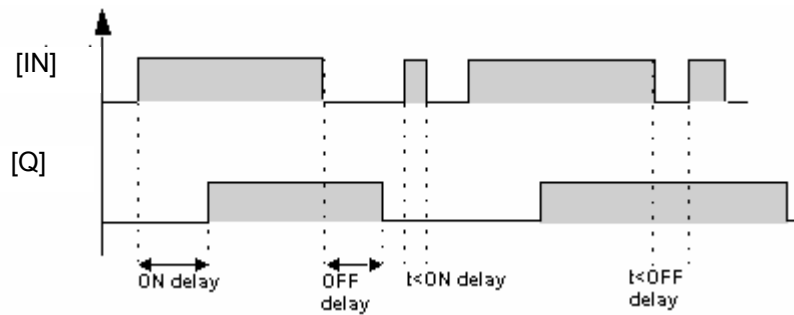
The following diagram shows the operation of the timer in function C:



Note: Each pulse on the Command input of the Timer block resets the current value to 0.

A/C Function

The following diagram shows the operation of the timer in function A/C:





ADD/SUB (ADD/SUB Arithmetic Function)

The ADD-SUB arithmetic function is used to perform simple operations on integers:

- Addition,
- Subtraction.

Calculation formula:

$$\text{CALCULATION OUTPUT [Q]} = \text{[A]} + \text{[B]} - \text{[C]}$$

Inputs/Outputs:

Description of the inputs:

- [A]: A signed input (S16: analog),
- [B]: A signed input (S16: analog),
- [C]: A signed input (S16: analog).

Note:

If the inputs are not connected, they are set to 0.

It is not possible to do mathematical operations on registers.

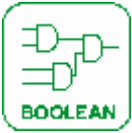
Description of the outputs:

- [Q]: Signed output (S16: analog): this is the value of the calculation formula output.
- [Overflow]: This output, whose type is Boolean, indicates any presence of saturation errors. This output is activated in the following cases.
 - Result not included in the interval [-32768, +32767]
 - The Error propagation input is active.

Examples:

Addition: do not use the Input [C].

Subtraction: do not use one of the [A] or [B].



Boolean Function

The BOOLEAN function gives the value of the output according to the combination of inputs. The function has four inputs, and therefore 16 combinations. These combinations can be found in a truth table; for each of these, the output value can be adjusted. The number of configurable combinations depends on the number of inputs connected to the function. Non-connected inputs are set to 0. The following diagram shows an example of part of the Boolean function truth table:

Inputs/Outputs:

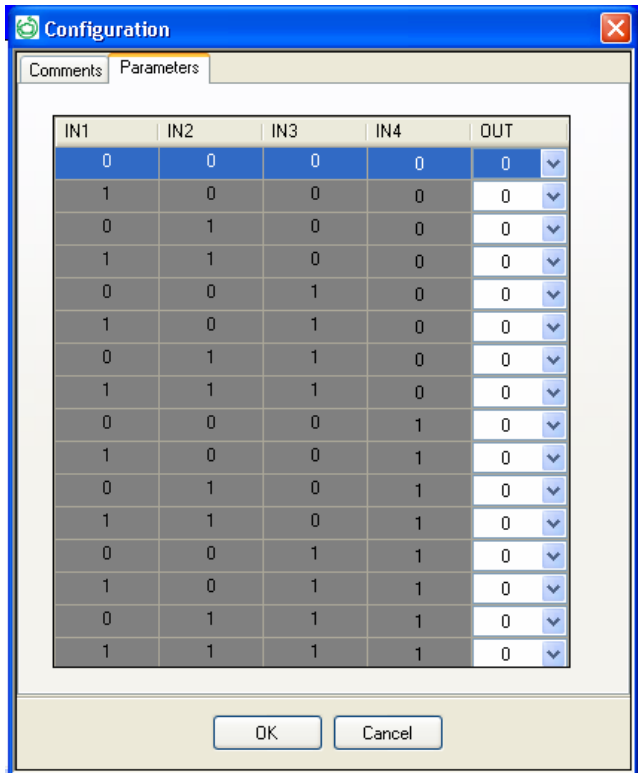
- [IN1], [IN2], [IN3] et [IN4]: 4 Boolean inputs.
- [OUT]: A Boolean output.

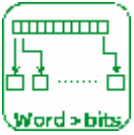
Configuration:

From the Configuration window, you can adjust in Parameters tab:

Having connected at least one input, configure the value of the output in the truth table, in the Parameters window.

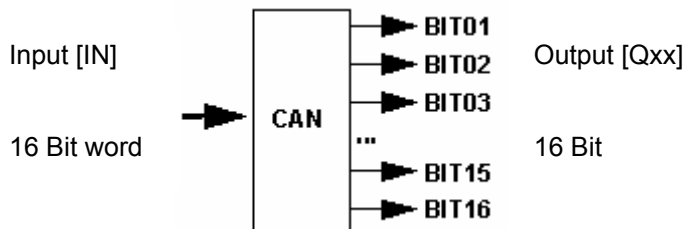
The output values can be 0 for the Inactive state, and 1 for the Active state.





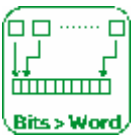
CAN (Word to Bit Conversion)

The Word to Bit Conversion function breaks down an integer-type input (16 bits) into 16 bit type outputs. ...



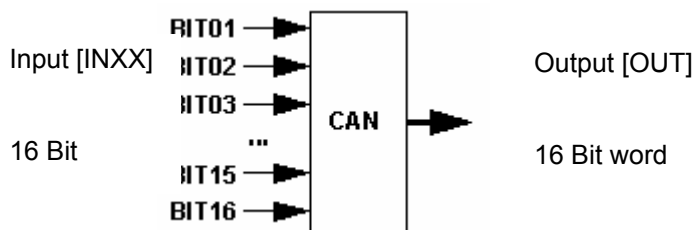
Inputs/Outputs:

- [IN]: A unsigned input (U16: register),
- [Q01] to [Q16]: 16 Boolean outputs: Bit01 (least significant byte) ... Bit16 (Most significant byte).



CNA (Bit to Word Conversion)

The Bit to Word Conversion function produces an integer (16 bits) type output from 16 bit-type inputs.



Inputs/Outputs:

- [IN01] to [IN10]: 16 Boolean entries: Bit01 (least significant byte) ... Bit16 (most significant byte),
- [OUT]: An unsigned output (U16: register).



COMPARE (Comparison of two values)

This function is used to compare two analog values.

The output [Q] is active if the result of the comparison between [IN1] and [IN2] is true and if the Enable functions input is active or not connected.

The output does not change state if the Enable function input changes from Active to Inactive state.

Inputs/Outputs:

- [IN1]: A signed input (S16: analog),
- [IN2]: A signed input (S16: analog).
- [ENABLE]: A Boolean input.

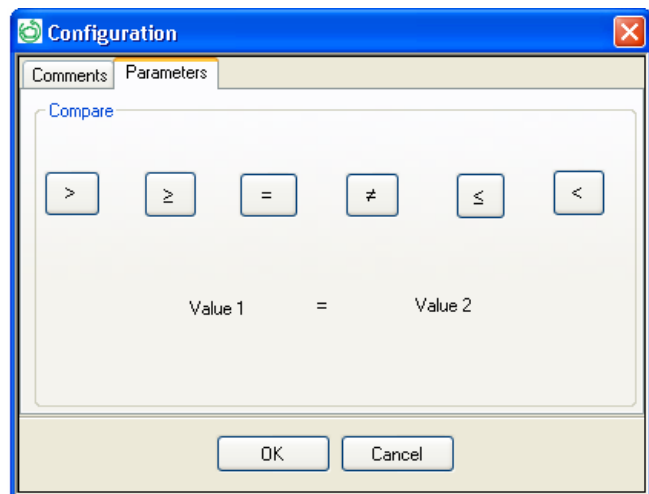
If the [IN1] or [IN2] input is not connected, the value is set to 0.

- [Q]: A Boolean Output.

Configuration:

From the Configuration window, in the parameters tab you can choose the comparison operators:

- > Greater than,
- ≥ Greater than or equal to,
- = Equal to,
- ≠ Different,
- ≤ Less than or equal to,
- < Less than.





Edge

The function detects falling or rising edge of a signal.

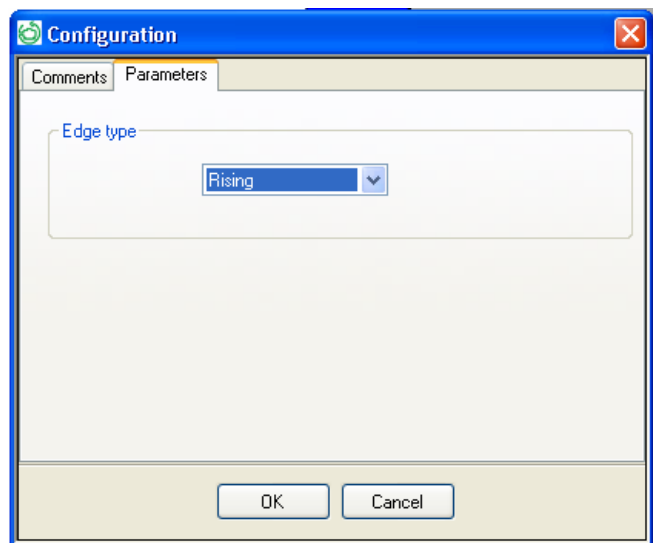
Inputs/Outputs:

- [IN]: A Boolean input.
 - Signal tested
- [Q]A Boolean Output.
 - Pulse when the edge is detected.

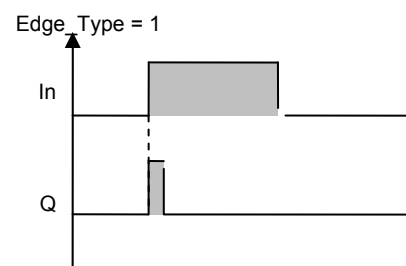
Configuration:

From the Configuration window, adjust in Parameters tab the Edge type:

- Falling
- Rising
- Twice



Select		In (t)	In(t+1)	Q (t)
Twice	Falling	1	0	1
		0	0	0
		1	1	0
		0	1	0
	Rising	1	0	0
		0	0	0
		1	1	0
		0	1	1





GAIN Function

The Gain function enables analog values to be converted by changing the scale and offset.

Gain calculation formula:

$$\text{CALCULATION OUTPUT} = (A / B) \times \text{CALCULATION INPUT} + C$$

Inputs description:

- [IN]: A signed input (S16: analog): This is an integer between -32768 and 32767.
- [ENABLE]: A Boolean input: This is the gain function input command.

The state of this input determines operation of the block: If the Enable Function input is inactive, the Calculation output retains the last calculated value.

Note: If the Enable Function input is not connected, it is considered to be active.

Outputs description:

- [Q]: A signed input (S16: analog): this is the output value of the gain function. This value depends upon the state of the Enable function input.
- [OVERFLOW]: A Boolean Output. If it is:
 - Inactive: the Calculation output is equal to zero,
 - Active: the Calculation output is equal to the result of the gain calculation formula.

Configuration:

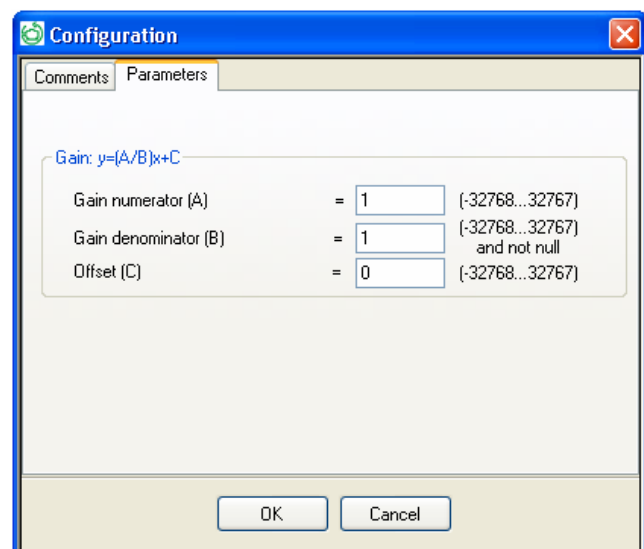
From the Configuration window, adjust in Parameters tab:

A/B which corresponds to the gain applied by the function with:

- A: being a numerator (from -32768 to 32767),
- B: being a denominator (from -32768 to -1 and from 1 to 32767) (4)
- C is the offset applied by the function, and is an integer between -32768 and 32767(5).

In addition, it is possible to define an operating range by setting limits for the function output:

Lower limit: integers between -32768 and 32767,
Upper limit: integers between -32768 and 32767.





MUL/DIV (MUL/DIV Arithmetic Function)

The MUL/DIV arithmetic function is used to perform operations on integers:

- Multiplication
- Division

Calculation formula:

$$\text{CALCULATION OUTPUT} = [A] \times [B] / [V]$$

Inputs description:

- [A]: A signed input (S16: analog),
- [B] A signed input (S16: analog),
- [C]: A signed input (S16: analog).

Note:

If the INPUTS are not connected, they are set to 1.

It is not possible to do mathematical operations on registers.

Outputs description:

- [Q]: A signed input (S16: analog): this is the value of the calculation formula output.
- [OVERFLOW]: This output, whose type is Boolean, indicates any presence of saturation errors. This output is activated in the following cases:
 - The consequence of the operations is a result that is not included in the interval [-32768, +32767],
 - The Error propagation input is active,
 - The Input 3 equals 0.

Examples:

Multiplication: do not use the [C] input.

Division: do not use one of the [A] Input or [B] inputs.



MUX (Multiplexing)

The Multiplexing function carries out two input channel multiplexing on the Output.

Inputs description

- [SEL]: A Boolean input: this input is used to choose the input channel to apply to the output.
- [IN1]: A signed input (S16: analog): this is the multiplexer input [IN1].
- [IN2]: A signed input (S16: analog): this is the multiplexer input [IN2]

Note:

If the Command input is not connected, it is considered to be inactive.

If channels [IN1] or [IN2] are not connected, they are set to 0.

Output description:

- [Q]: A signed output (S16: analog): this is the multiplexer output [Q].

This value depends upon the state of the Command input [SEL].

If the Command input is:

- Inactive: the Output corresponds to [IN1].
- Active: the Output corresponds to [IN2].



ReadBitParam

The “**ReadBitParam**” function block is used for reading one bit into parameters.

To read one bit of drive parameters:

- Use the ADL containers to link with the parameters of the drive.
(See description of the [\[ADL Containers\] \(LA01\)](#))

To read one bit of parameters of a communication protocol:

- Use the parameters M00x to store data. Store values for which they are reused by the function blocks and communication protocol.
(See description of the [\[FB PARAMETERS\] \(M001\)](#))

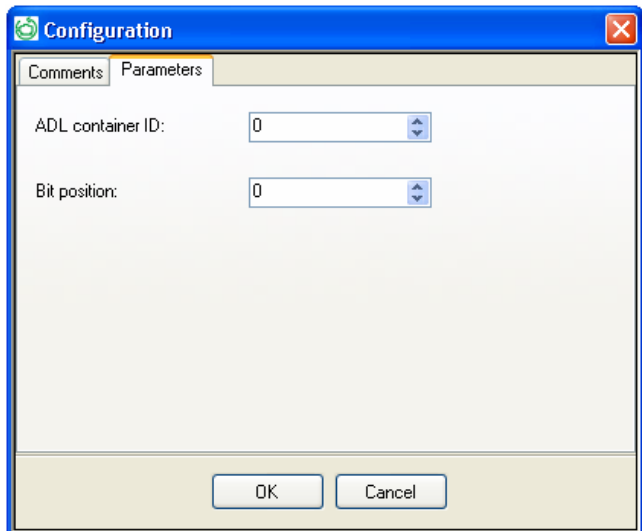
Inputs/Outputs

- [ENABLE]: A Boolean input,
- [Q]: A Boolean ReadBit output.

Configuration

From the Configuration window, you can adjust in Parameters tab:

The ADL Container ID will be used and choose the Bit position.





ReadParam

The “**ReadParam**” function block is used for read a value.

To read drive parameters:

- Use the ADL containers to link with the parameters of the drive. (See description of the [\[ADL Containers\] \(LA01\)](#))

To read parameters of a communication protocol:

- Use the parameters M00x to store data. Store values for which they are reused it by the functions blocks and communication protocol. (See description of the [\[FB PARAMETERS\] \(M001\)](#))

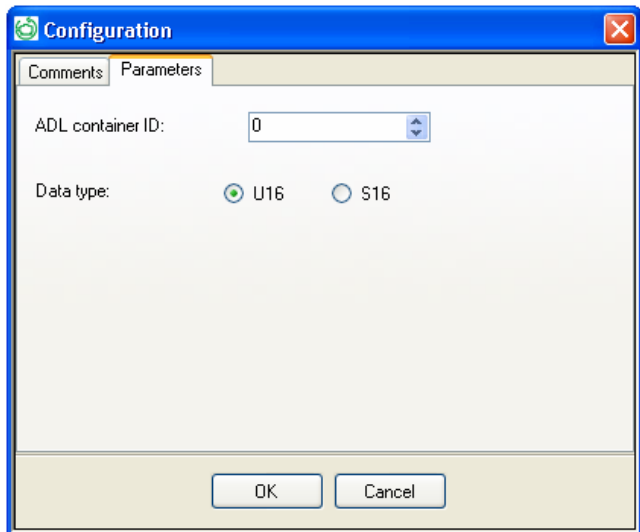
Inputs/Outputs

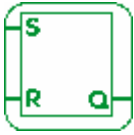
- [ENABLE]: A Boolean input.
- [Q_U16]: An unsigned output (U16: register).
- [Q_S16]: A signed output (S16: analog),

Configuration

From the Configuration window, you can adjust in Parameters tab:

The ADL Container ID will be used. You can also select the type of output signal.





RS memory

The SET RESET function operates as follows:

- Activation of the SET input activates the output, which remains so even if the SET input is then deactivated,
- Activation of the RESET input deactivates the output.

Non-connected inputs are set to the Inactive state.

Inputs/Outputs

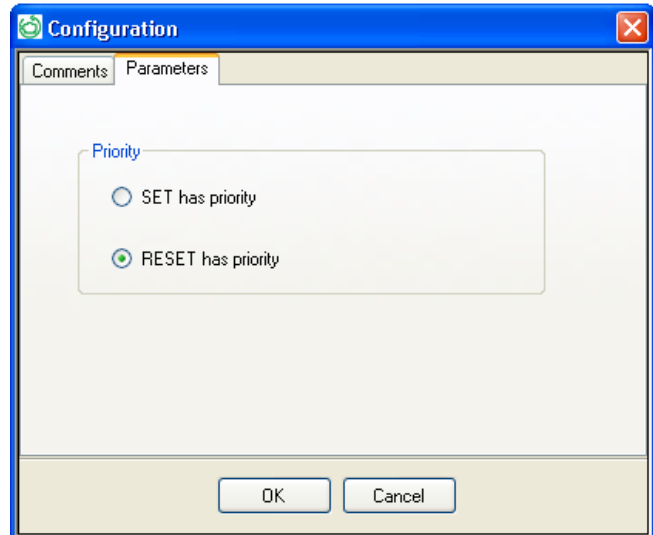
- [SET]: A Boolean input,
- [RESET]: A Boolean input,
- [Q]: A Boolean output.

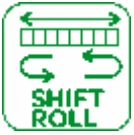
Configuration:

From the Configuration window, you can adjust in Parameters tab:

If both inputs are active, the state of the output depends on the configuration of the function:

- The [Q] output is active if the [SET] has priority option is configured,
- The [Q] output is inactive if the [RESET] has priority option is configured.





Shift/Roll

The function allows to shift or rolls to the left/right, the [IN] value of a fixed number of bits.

At each cycle if the [Enable] signal is set, the logic block will perform a shift/roll to the left/right to the Number Bit request regarding the parameters configuration.

SHIFT move bits in the selected direction and replace empty bits by 0.

Note: It is impossible to lag records on analog values.

Inputs description:

- [IN]: An unsigned input (U16: register).
- [ENABLE]: A Boolean function input,

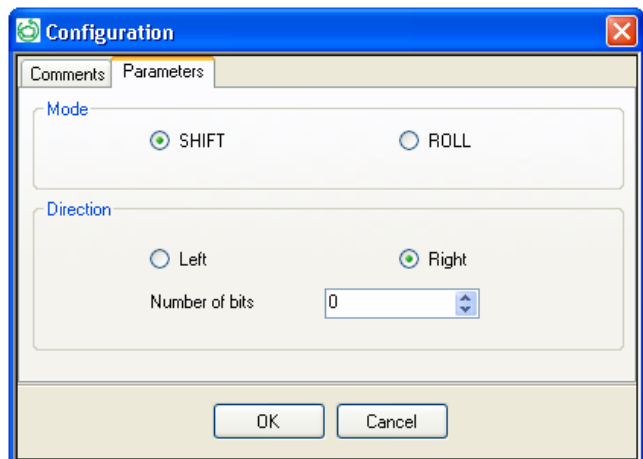
Outputs description:

- [Q]: A unsigned output (U16: analog),
 - [CARRY]: A Boolean output.
- The carry output tells you the value of the last bit shifted.

Configuration:

From the Configuration window, you can adjust in Parameters tab:

- The mode: Shift or Roll.
- The direction: Left or right.
- The number of bits.





TRIGGER (Schmitt Trigger)

The Schmitt Trigger function allows an analog value to be monitored relative to two thresholds. The output changes state if:

- The [IN] value is less than the minimum value,
- The [IN] value is greater than the maximum value.

If the [IN] input is between the two, the [Q] output does not change state.

Each of from On to Off and from Off to On setpoints can be set as the minimum or maximum value. This involves reverse operation of the function. These two operations are shown in the diagrams.

If the [Enable] input is in inactive state, the [Q] output remains inactive. The [Q] output does not change state if the [Enable] input changes from Active to Inactive state.

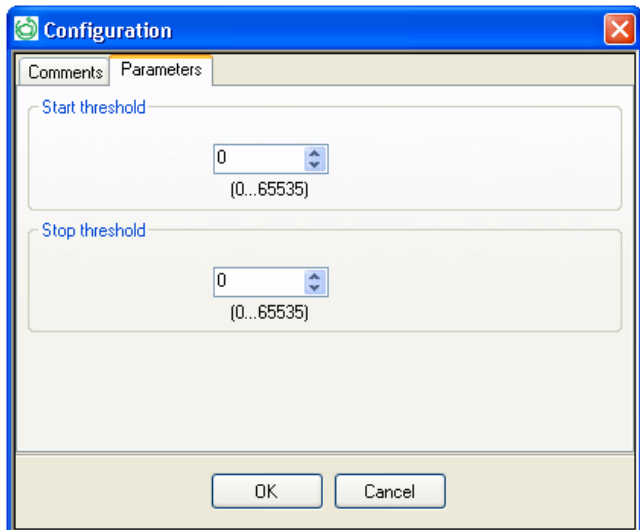
Inputs/Outputs:

- [IN]: An Integer Off and On setpoint input,
- [ENABLE]: A Boolean input
- .
- [Q]: A Boolean Output.

Configuration:

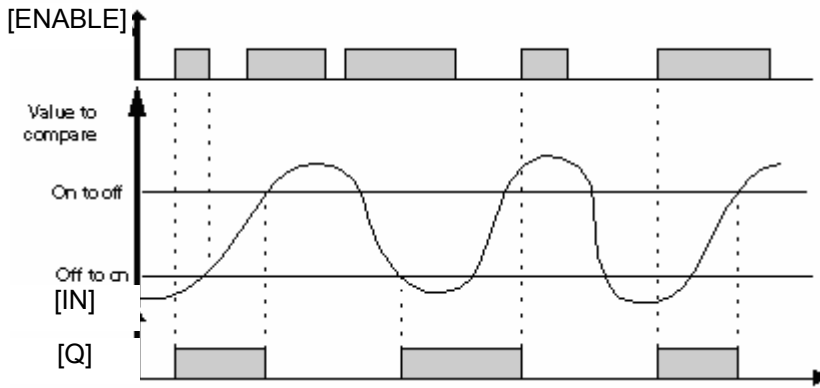
From the Configuration window, you can adjust in Parameters tab:

- Start threshold,
- Stop threshold.

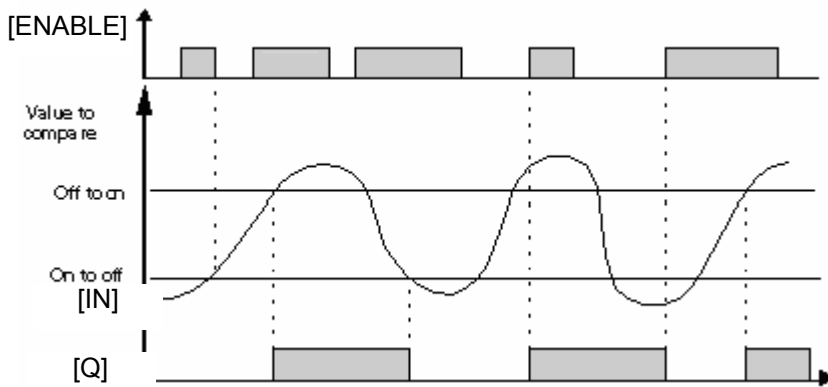


Operating Diagrams:

The figure shows possible output states when the On to Off setpoint is higher than the Off to on setpoint:



The figure shows the possible output states when the Off to on setpoint is higher than the On to Off setpoint:





UP/DOWN and PRESET COUNT

The Up/Down Counter function is used to up-count or down-count from a preset value resulting from a calculation outside the function.

Inputs:

- [UP]: A Boolean Up-Count input,
- [DOWN]: A Boolean Down-Count input,
- [RESET]: A Boolean input,
- [LOAD]: A Boolean input,
- [PV]: An analog Preset value input.
- [ENABLE]: An input,

Outputs:

- [Q]: A Boolean Output,
- [CURRENT]: The current counter value.

Several functions are available:

- Up-counting and forcing the counter to 0 on initialization,
- Up-counting and forcing the counter to 0 on initialization and when the count value has been reached,
- Down-counting and forcing the counter to the preset value on initialization,
- Down-counting and forcing the counter to the preset value on initialization and when 0 has been reached.

Activation to 1 of the [LOAD] input is used to change the counter with the value available at the [PV] input.

The [PV] input can be connected to an NUM constant, to an analog input, or to any other kind of function block output which delivers an signed-type value.

A rising edge on the:

- Upcounting: increments the counter.
- Downcounting: decrements the counter.

State of the Output:

Is at 1: When the counting number has been reached, the [Q] output switches to 1 and remains so for as long as the counting number is greater than or equal to the [PV] Preset value,

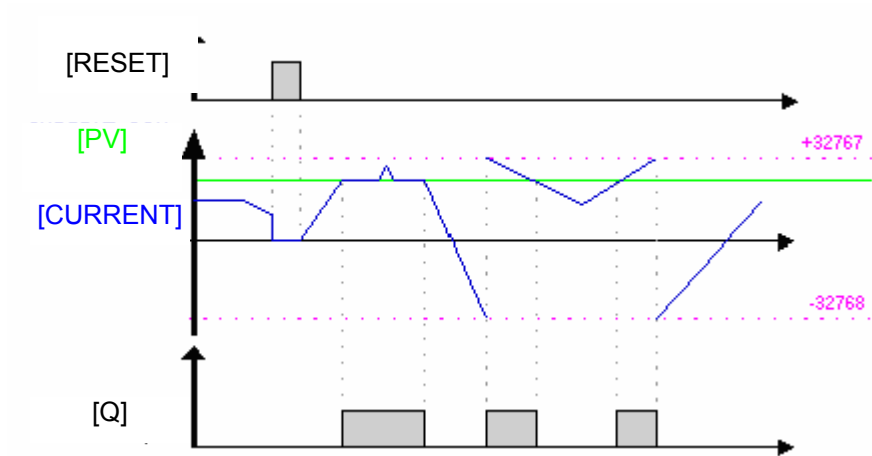
Is at 0: If the transitions on the Downcounting input switch the counting number back to a value less than the Preset value.

Activation of the [Reset] or [LOAD] inputs enables the counter to be reloaded.

When the [Reset] input is set to 1, the [Q] output remains in state 0. When the [Reset] input becomes 0, the up/down counting operation is restarted from zero.

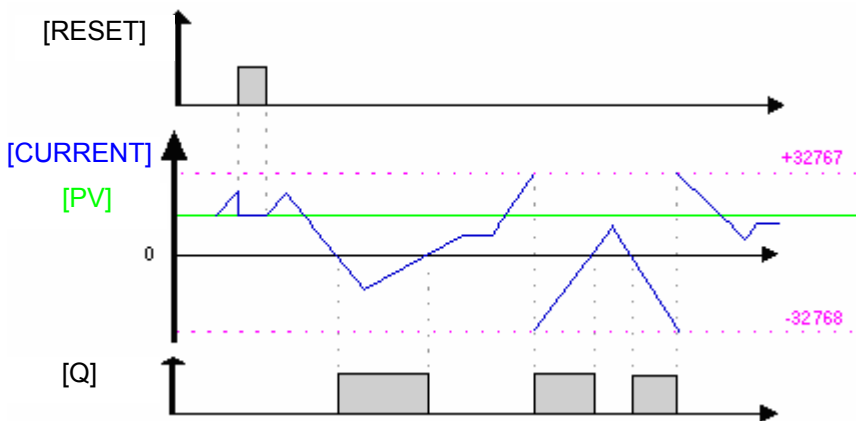
For the following four charts, the blue curve represents the internal counter value, when it increases there are pulses on the upcount input and when it decreases, pulses on the downcount input. The following diagram shows the operation of the counter with initialization at 0:

Up-Counting in Single Cycle Mode:



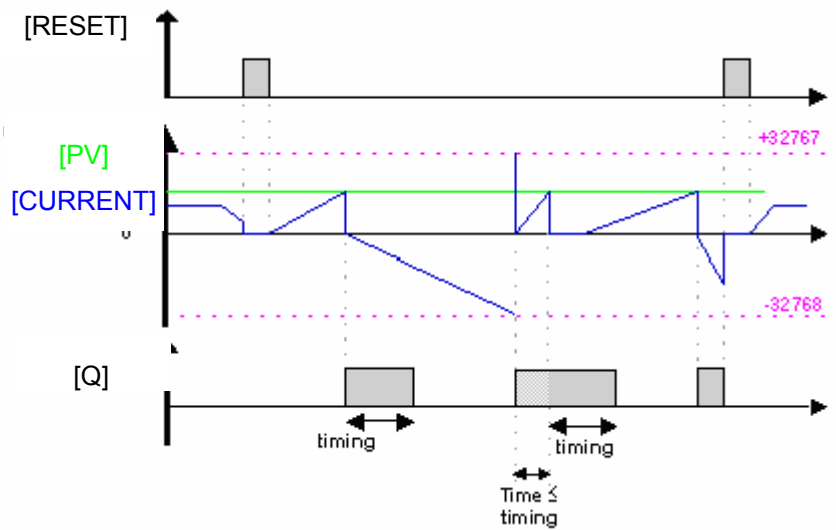
Down-Counting in Single Cycle Mode:

The following diagram shows the operation of the down-counter with initialization at the preset value:



Up-counting in Repetitive Cycle Mode:

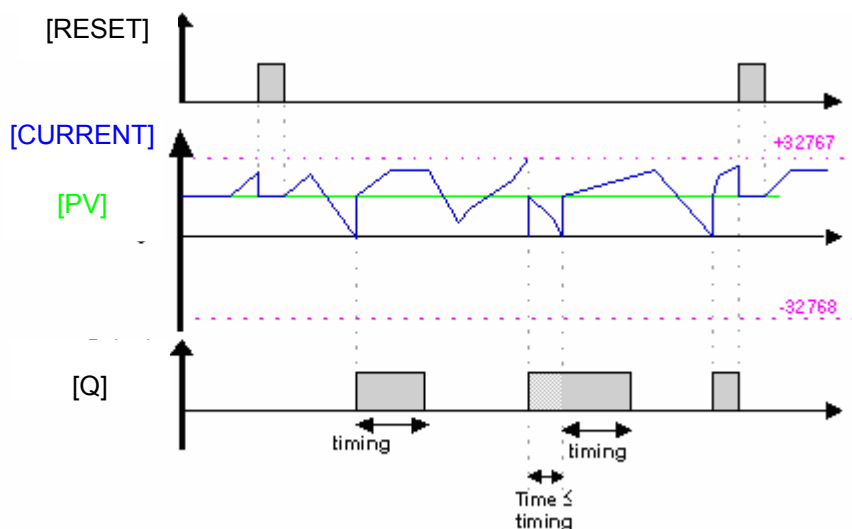
The following diagram shows the operation of the counter with forcing to 0 of the [CURRENT] current value on [RESET] initialization, or when the count value has reached the [PV] preset value:



The [Q] output switches to the Inactive state when the predefined pulse duration value has run out. If the switch condition is Active before switching to the Inactive state, the output pulse is extended by the Duration of pulse (Timing).

Down-Counting Function in Repetitive Cycle Mode:

The following diagram shows the operation of the down-counter with forcing to the [PV] preset value of the [CURRENT] current value on initialization, or when the count value has reached 0:



The [Q] output switches to the Inactive state when the predefined pulse duration value has run out. If the switch condition is Active before the switch to the Inactive state, the output pulse is extended by the Duration of pulse (Timing).



WriteBitParam

Description

The “**WriteBitParam**” function block is using for write one Bit into parameters.

For write one Bit of drive parameters:

You must use the ADL containers to link with the parameters of the drive. (See description of the [\[ADL CONTAINERS\]](#) (LA01))

Note:

When a backup in EEPROM is being, the “WriteBitParam” are no longer operational.

For store or reused a value:

The parameters M00x are easy to use to store data. You can store values for which they are reused it by the functions blocks and communication protocol. (See description of the [\[FB PARAMETERS\]](#) (M001))

Inputs/Outputs

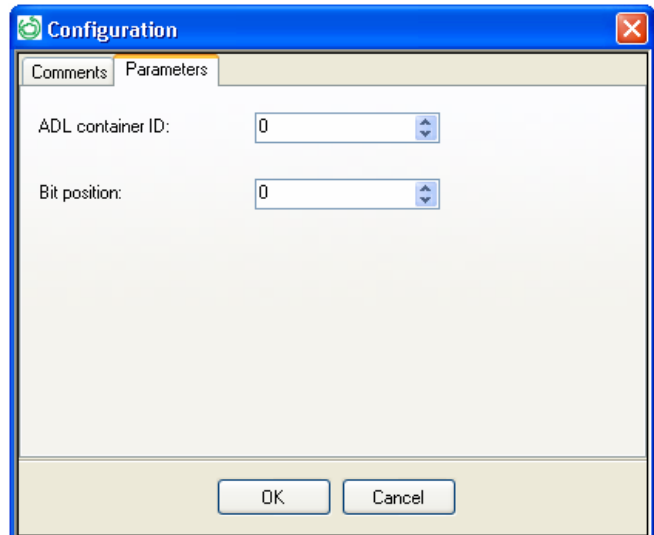
The WriteBitParam uses:

- [IN]: A Boolean WriteBit input,
- [ENABLE]: A Boolean input.

Configuration:

From the Configuration window, you can adjust in Parameters tab:

In the parameters tab you can choose the ADL Container ID will be used and choose of Bit position.





WriteParam

Description

The “**WriteParam**” function block is use for write or store a value.

For write drive parameters:

You must use the ADL containers to link with the parameters of the drive. (See description of the [\[ADL Containers\]](#) (LA01))

Note:

When a backup in EEPROM is being, the “WriteParam” are no longer operational.

For store or reused a value:

The parameters M00x are easy to use to store data. You can store values for which they are reused it by the functions blocks and communication protocol. (See description of the [\[FB PARAMETERS\]](#) (M001))

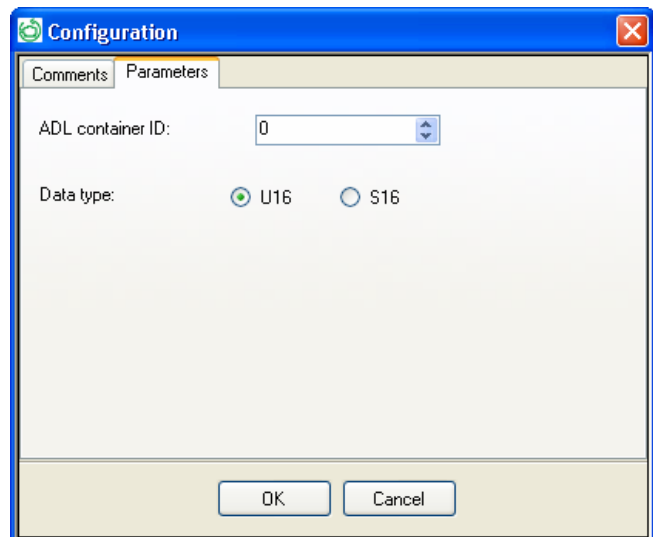
Inputs/Outputs

- [IN_U16]: An unsigned input (U16: register).
- [IN_S16]: A signed input (S16: analog),
- [ENABLE]: A Boolean input.

Configuration:

From the Configuration window, you can adjust in Parameters tab:

In the comments tab you can choose the ADL Container ID will be used. You can also select the type of input signal.





STU (Signed 16bits TO Unsigned 16Bits)

Description

The “**STU**” function block is use for convert a signed 16 Bits to an unsigned 16 Bits.

Signed input range: -32768 to +32767.

Unsigned output range: 0 to 32767.

Inputs/Outputs

➤ [IN]: A signed input (S16: analog).

➤ [Q]: An unsigned output (U16: register),

➤ [OVERFLOW]: An Overflow output.

The Overflow output is of Boolean type, this output is set if, the input signed value is negative, the output unsigned value will be set to 0 and the Overflow.



UTS (Unsigned 16bits TO Signed 16Bits)

Description

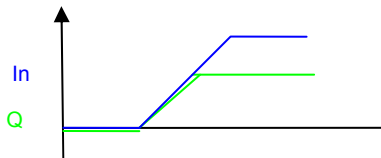
The “**UTS**” function block is use for convert an unsigned 16 Bits to a signed 16 Bits.

Unsigned input range: 0 to 65535.

Signed output range: -32768 to +32767.

Inputs/Outputs

- [IN]: An unsigned input (U16: register).
- [Q]: A signed output (S16: analog),
- [OVERFLOW]: An Overflow output.
The Overflow output is of Boolean type, this output is set if, the input unsigned value is over 32767, the output signed value will be set to 32767.



Communication option

Configuration of connection

To establish communication between the Function Blocks via SoMove and the ATV 32, one of the following links may be used:

- Serial link: Com port
- Bluetooth link: Com port

Before launching the connection between Function Blocks and the ATV 32, the following must be checked:

In the case of...	Make sure that:
Serial link	The smart relay is physically connected to the SoMove (PC), The connection is correctly configured.
Bluetooth link	The bluetooth of the ATV 32 is activated from the parametre [Bluetooth Activation] (btUA) = [ON] (On) . Note: The driver associates a com port to the adapter.

Program Import

The **File → Import** command and  is used to import a program

An entire program,

- One or several contiguous lines,
- One or several contiguous cells.

Following each import, an automatic program check is performed.

Following each import, you are advised to launch the **Edit → Check the program command**.

Note: It is possible to import a subset of an FBD program using copy and paste.

Parameters of communication

Description of ADL Containers

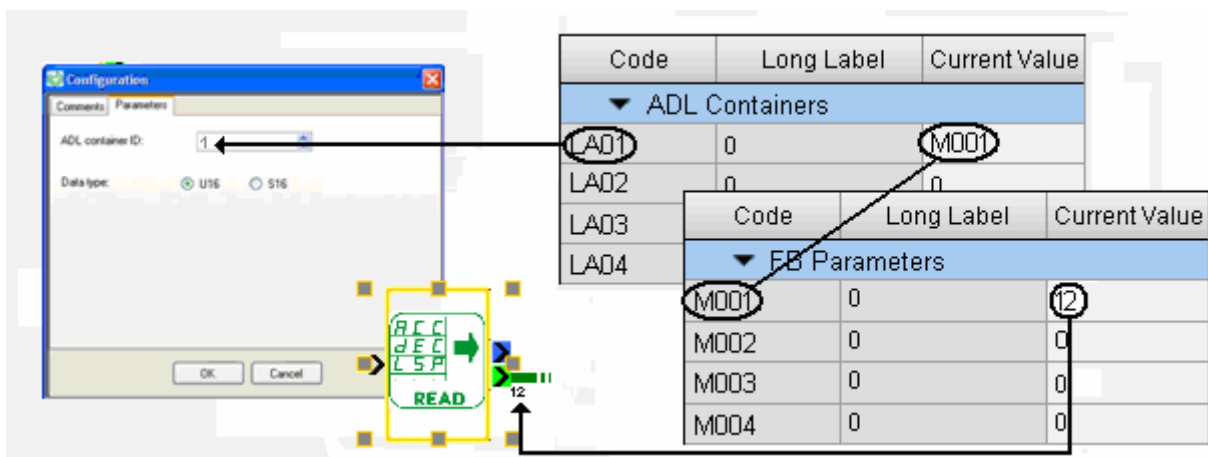
The ADL Containers are linked to 4 function blocks

- ReadParam,
- WriteParam,
- ReadBitParam,
- WriteBitParam.

Because, those function blocks are linked to ADL container (by configuring a blocks parameter name "Container ID"). This configuration is done during graphical edition of the program in Function Blocks.

There are 8 ADL containers used to connect %MW, %SW, or other drive parameters.

Ajouter un lien vers le guide des paramètres ADLC



Depending on which task the Read/WriteParam logic block is implemented, it will allow connection to Fast or Slow parameters.

The rule is:

- In PRE/POST tasks: Connection to Fast access parameters only.
- In AUX task: Connection to Slow and Fast access parameters.

4 groups of parameters are defining:

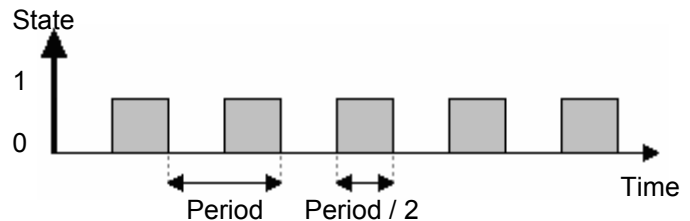
- FB Read fast (for PRE/POST tasks)
- FB Read slow (for AUX task)
- FB Write fast (for PRE/POST tasks)
- FB Write slow (for AUX task)

FB SYSTEM WORDS

[FB SYSTEM WORD 01] (S001)							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TB1MIN : Clock bit : period 1min, duty cycle 50%	TB1SEC : Clock bit : period 1s, duty cycle 50%	TB100MS : Clock bit : period 100ms, duty cycle 50%	Reserved	Reserved	Reserved	Reserved	Reserved
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reserved	Reserved	1RSTSCANRUN : FB is executing its 1rst cycle since last RUN. Used only in PRE and POST	Reserved	Reserved	Reserved	Reserved	Reserved

Bits 5, 6 and 7 description:

An internal timer regulates the change of states of the bit.



Bits 13 description:

The portion of mode STOP to RUN mode (even after a cold start) is reported by setting 1 of %S13 system bit. This bit is reset to 0 at the end of the first cycle of the POST task in run mode.

[FB SYSTEM WORD 06] (S006)							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reserved	Reserved	USRMWSAVE : Users variable save request (=CMI bit 15)	Reserved	Reserved	Reserved	Reserved	Reserved

Bits 13 description:

The bit %S13 is the bit %S94 on Modicon M340 and Premium.

When a backup in EEPROM is being, the writes are no longer operational.

Supervision/Monitoring Window

The supervision/monitoring window is a subset of the edit window. .

It can be accessed from:

Simulation: the Mode/Simulation menu or using the simulation button on the toolbar

Monitoring: the Mode/Monitoring menu or using the simulation button on the toolbar

It contains the functions, without their connections, that the programmer extracted (using Drag/Drop or Copy/Paste) from the edit window.

In simulation and monitoring mode, the parameters and outputs of the functions present are updated.