User manual

Configuring the calculation module

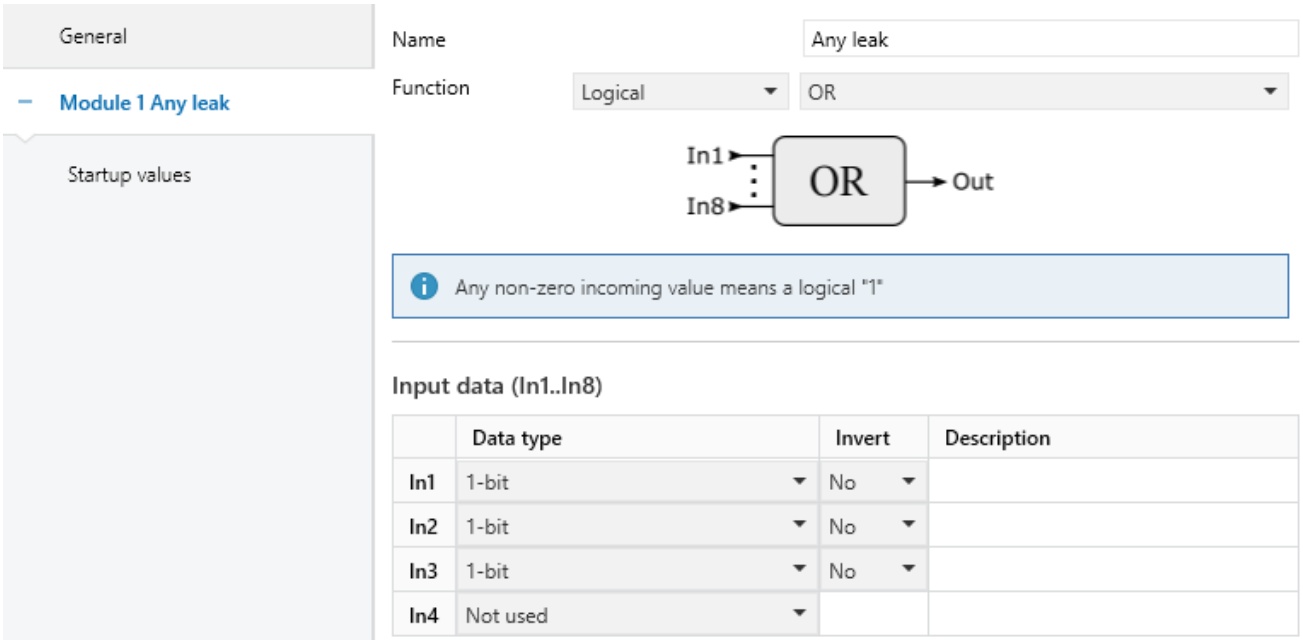
Version 1.0

Contents

1. General

All functions of the calculation module are divided according to the criterion of the data type of the intermediate value (the result of the actual function) into two main types:

1) Functions with a binary intermediate data type (these are all "Logical functions", all "Comparison functions", as well as from the other category: "Logical conversion / Gate", "Switch with 2 thresholds", "Staircase timer", "Differential switch". The output stage of these functions is configured uniformly: description in Chapter 7
2) Functions with a numeric intermediate data type (these are all "Mathematical functions", "Statistical functions", as well as from the other category: "Format conversion / Gate", "Pulse counter", "Multiplexer", "Input number indicator", "Memory cell" The output stage of these functions is configured uniformly: description in Chapter 8



«Name»
Possible values: String up to 20 characters long

The name of the channel is displayed in the header of the settings tab and the communication objects corresponding to the channel

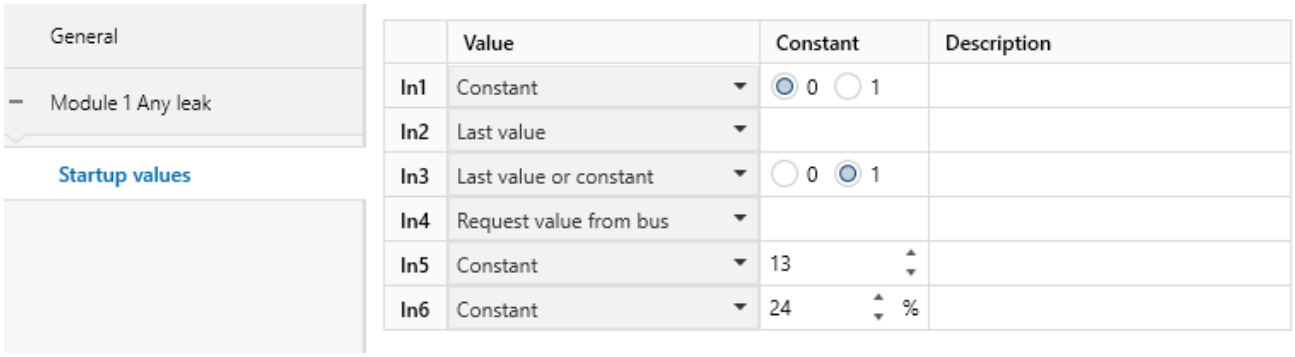
«Function»
Selecting the module function. All functions are divided into 5 groups according to their types.

Function group, Possible values: {«Logical», «Mathematical», «Comparators», «Statistical», «Other»}

- **Logical**
  - OR
  - AND
  - XOR
  - Multiplexer 1bit
  - Memory cell, 1bit
- **Mathematical**
  - Addition
  - Subtraction
  - Division
  - Multiplication
  - Reminder of division
  - Accumulator

- **Comparators**
  - All equal
  - First smallest
  - First largest
  - First smallest or equal
  - First largest or equal
- **Statistical**
  - Maximum
  - Minimum
  - Average
  - Trimmed mean
  - Moving average
  - Number of non-zero values
- **Other**
  - Logical conversion / Gate
  - Format conversion / Gate
  - Switch with 2 thresholds
  - Differential switch
  - Pulse counter
  - Multiplexer
  - Input number indicator
  - Memory cell
  - Staircase timer

1.1. Startup values

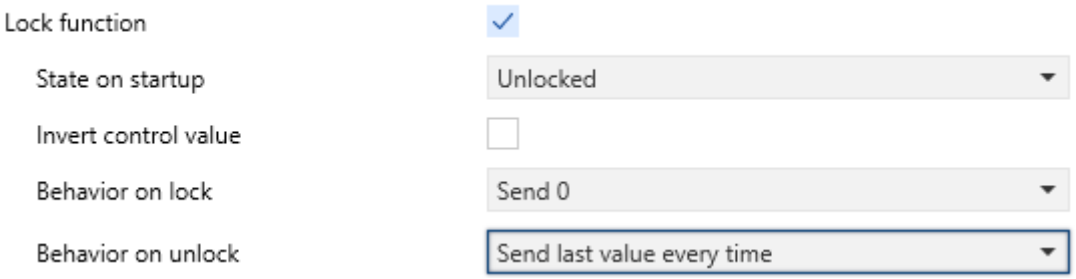


In all functions, a default value can be defined for all defined inputs (except constants), which is used for input when the device starts and before the first value arrives from the KNX bus. One of five configuration options can be used:

1) "Undefined" - the input value is undefined until the first value arrives from the KNX bus. Until all inputs have a certain moment value, the module will act and will not transmit any values to the output.
2) "Constant" - specifies the value for the input that will be effective upon the first arrival of the value from the KNX data bus. The data type corresponds to the data type assigned to the corresponding input.
3) "Last value or constant" - if there is a previously saved value from the data bus, it is used for the default value, otherwise the specified constant
4) "Last value" - if there is a previously saved value from the data bus, it is used for the default value, otherwise the input value is not defined, and the module is not valid until all inputs receive at least some value.
5) "Request value from bus" - when the device is started, it requests a value from the data bus, in the absence of a response value, the input value remains in an undefined state, and the module, respectively, in an inactive state.

1.2. Blocking module execution

Each calculation module has a special block for configuring the module execution lock. A module in a locked state does not change the output value with any change in the input values.



«Lock function»
Possible values: {Off/On}

If the value is "On": enabling the functionality of blocking the execution of the calculation module, enabling the communication object "Lock"

«State on startup»
Possible values: {«Unlocked», «Locked», «Last state»}

The setting determines the lock status of the module when the device is turned on.

«Invert control value»
Possible values: {Off/On}

With the value "Off", the lock is carried out with the value "1" on the communication object "Lock"
With the value "On", the lock is carried out with the value "0" on the communication object "Lock".

«Behavior on lock»
Possible values: {«None», «Send last value once», «Send last value every time», «Send 0»}

The setting determines the behavior of the module at the time of the arrival of the lock command in the "Lock" communication object (considering the "Invert control value" parameter)

- «None» - it does not give any values
- « Send last value once» - if the module is in the active state, it outputs the last calculated value, when blocking commands are received on an already blocked module, it no longer outputs any values.
- «Send last value every time » each time the lock command arrives, the module outputs the last value calculated before the first lock
- «Send 0» - when locked, the module outputs either "0" for numeric type operations or a value corresponding to a logical 0 for logical type operations.

«Behavior on unlock»
Possible values: {«None», «Send last value once», «Send last value every time»}

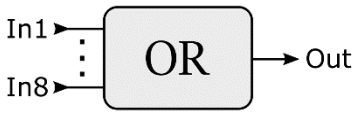
The setting determines the behavior of the module at the time of the arrival of the unlock command in the "Lock" communication object (considering the "Invert control value" parameter)

- «None» - it does not give any values
- « Send last value once» - the module issues the last value calculated before the module was blocked on a one-time basis, the receipt of subsequent commands to unlock does not initiate any actions of the module.
- «Send last value every time» - with each unlock command, the module outputs the last calculated value: when unlocking, the last calculated value before locking, in the already active state, initiates a repeat of the last calculated value.

2. Logical functions

2.1. Logical «OR»

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out



The module calculates an intermediate value "O" (0 or 1) – depending on which it outputs one of two preset values of the selected type to the output of the module. The module is active only if all the configured inputs have a certain value.

The intermediate value "O" is set to "1" if at least one input (considering the inversion setting) is different from 0, otherwise the intermediate value is set to "0"

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

2.2. Logical «AND»

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out

The module calculates an intermediate value "O" (0 or 1) – depending on which it outputs one of two preset values of the selected type to the output of the module. The module is active only if all the configured inputs have a certain value.

The intermediate value "O" is set to "1" if the values of all inputs are different (considering the inversion) from 0, otherwise the intermediate value is set to "0"

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

2.3. Logical «XOR»

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out

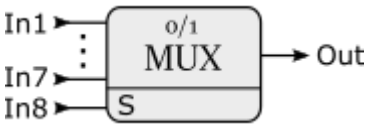
The module calculates an intermediate value "O" (0 or 1) – depending on which it outputs one of two preset values of the selected type to the output of the module. The module is active only if all the configured inputs have a certain value.

The intermediate value "O" is set to "1" if an odd number of inputs (considering the inversion setting) have a value other than 0, otherwise the intermediate value is set to "0"

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

2.4. «Multiplexer, 1bit»

Inputs:
Data: {In1…In7}
Selector In8(S)
Output:
O ⇒ Out

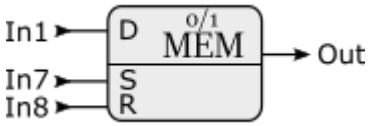


The intermediate value "O" is set to "1" if the input value (considering the inversion) of the selected input value In8(S) is different from zero and "0" otherwise. When the input value is In8(S)=0, input In1 is selected as the data source, when In8(S)=1, In2 is selected, and so on. There is no module execution result until the first value arrives at the In8(S) input or the input selected by it.

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

2.5. «Memory cell, 1bit»

Inputs:
Data: In1 (D)
Store: In7 (S)
Read: In8 (R)
Output:
O ⇒ Out



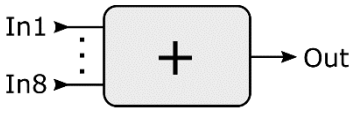
When "1" enters the input In7(S), a value equal to "1" is stored in the element's memory if the last value that came to the input In1(D) is different from "0" and "0" otherwise. When a "1" is received at the In8(R) input, the stored value ("0" or "1") is further transmitted to processing as an intermediate value "O"

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

3. Mathematical functions

3.1. Summation

Inputs:
Data: {In1…In8}
Outputs:
O ⇒ Out
Error (DPT_Alarm)



The module calculates the sum of all connected inputs {In1...In8} and uses it as an intermediate value of "O". The module performs the operation only if all the configured inputs have a certain value.

$$O = In1 + In2 + In3 + In4 + In5 + In6 + In7 + In8$$

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

3.2. Subtraction

Inputs:
Data: {In1…In8}
Outputs:
O ⇒ Out
Error (DPT_Alarm)

The module calculates the difference between the values of In1 and the sum of {In2...In8} and outputs it to the intermediate value "O". Only configured inputs are used in the calculation. The module performs the operation only if all the configured inputs have a certain value.

$$O = In1 - (In2 + In3 + In4 + In5 + In6 + In7 + In8)$$

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

3.3. Division

Inputs:
Data: In1, In2
Outputs:
O ⇒ Out
Error (DPT_Alarm)

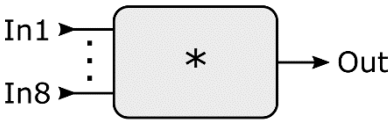
The module calculates the result from dividing In1 by In2 and outputs it in "O". In case of an attempt to divide by 0, the additional Communication object «Error» is set to the value "1".

O = In1 / In2

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

3.4. Multiplication

Inputs:
Data: {In1…In8}
Outputs:
O ⇒ Out
Error (DPT_Alarm)



The module calculates the result of multiplying all incoming values In1...In8 and outputs it in "O". The module performs the operation only if all the configured inputs have a certain value.

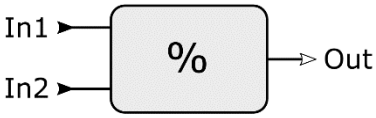
O = In1 * In2 * In3 * In4 * In5 * In6 * In7 * In8

In case of an overflow of the result of the operation, bit "1" is set in the additional Communication object "Error" for the data type defined as the output

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

3.5. Remainder of division

Inputs:
Data: In1, In2
Outputs:
O ⇒ Out
Error (DPT_Alarm)



The module calculates the remainder of the division of In1 by In2 and outputs it to the intermediate value "O".

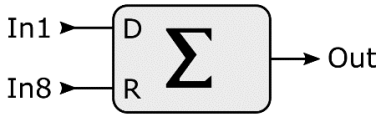
O = Mod [In1/In2]

In case of an attempt to divide by 0, the additional Communication object "Error" is set to the value "1".

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

3.6. Data accumulator

Inputs:
Data: In1 (D)
Reset: In8 (R)
Outputs:
Out
Error (DPT_Alarm)



Parameters:

«Start value»
Possible values: {-1000000.00…1000000.00}

The floating-point number from which the accumulation will be carried out and to which the reset will be carried out.

«Reset when threshold is reached»
Possible values: {Off/On}

When set to "On" and the accumulated amount reaches the maximum possible value for the selected data type, the module value will be automatically reset to the Initial value

The module calculates the sum of the sequence of incoming In1 values and outputs them in "O" when each value arrives at the In1(D) input to move the value of the module to the initial one, it is necessary to apply "1" to In8(R).

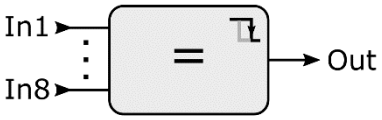
$$O = In1(t1) + In1(t2) + \ldots + In1(tN)$$

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

4. Comparators

4.1. All equal

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out



Parameters:
«With upper and lower hysteresis»
Possible values: {Off/On}.

When this option is selected, the comparison operation is performed considering the upper and/or lower hysteresis. Otherwise, the comparison is made for an exact match.

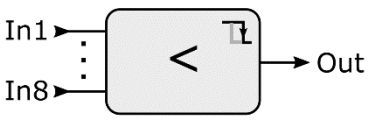
«Upper hysteresis»
Possible values: {0.00…1000000.00} (floating point number)
«Lower hysteresis»
Possible values: {-1000000.00…0.00} (floating point number)

The intermediate output value of "O" is "1" if and only if all the values at the inputs are In1...In8 are equal. Otherwise, "O" is equal to "0". When the hysteresis is on, the switching points from "0" to "1" and back are shifted according to the hysteresis values.

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

4.2. First smallest

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out



Parameters:
«With upper and lower hysteresis»
Possible values: {Off/On}.

When this option is selected, the comparison operation is performed considering the upper and/or lower hysteresis. Otherwise, the comparison is made for an exact match.

«Upper hysteresis»
Possible values: {0.00…1000000.00} (floating point number)

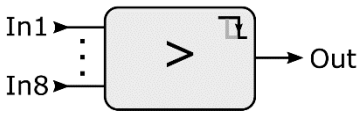
«Lower hysteresis»
Possible values: {-1000000.00…0.00} (floating point number)

The intermediate output value of "O" is "1" when In1 is less than any value at the inputs In2...In8. Otherwise, "O" is "0". When the hysteresis is on, the switching points from "0" to "1" and back are shifted according to the hysteresis values.

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

4.3. First largest

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out



Parameters:
«With upper and lower hysteresis»
Possible values: {Off/On}

When this option is selected, the comparison operation is performed considering the upper and/or lower hysteresis. Otherwise, the comparison is made for an exact match.

«Upper hysteresis»
Possible values: {0.00…1000000.00} (floating-point number)

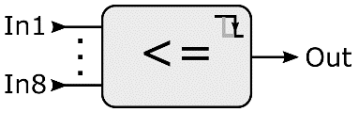
«Lower hysteresis»
Possible values: {-1000000.00…0.00} (floating-point number)

The intermediate value of "O" is equal to "1" when In1 is greater than any value at the inputs In2...In8. Otherwise, "O" is equal to "0". When the hysteresis is on, the switching points from "0" to "1" and back are shifted according to the hysteresis values.

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

4.4. First smallest or equal

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out



Parameters:
«With upper and lower hysteresis»
Possible values: {Off/On}

When this option is selected, the comparison operation is performed considering the upper and/or lower hysteresis. Otherwise, the comparison is made for an exact match.

«Upper hysteresis»
Possible values: {0.00…1000000.00} (floating point number)

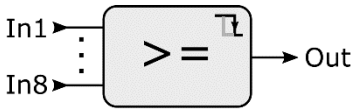
«Lower hysteresis»
Possible values: {-1000000.00…0.00} (floating point number)

The intermediate value of "O" is equal to "1" if and only if In1 is less than or equal to any value at the inputs In2...In8. Otherwise, "O" is equal to "0". When the hysteresis is on, the switching points from "0" to "1" and back are shifted according to the hysteresis values.

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

4.5. First largest or equal

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out



Parameters:
«With upper and lower hysteresis»
Possible values: {Off/On}.

When this option is selected, the comparison operation is performed considering the upper and/or lower hysteresis. Otherwise, the comparison is made for an exact match.

«Upper hysteresis»
Possible values: {0.00…1000000.00} (floating point number)
«Lower hysteresis»
Possible values: {-1000000.00…0.00} (floating point number)

The output value of "O" is "1" if and only if In1 is greater than or equal to any value at the inputs In2...In8. Otherwise, "O" is "0". When the hysteresis is on, the switching points from "0" to "1" and back are shifted according to the hysteresis values.

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

5. Statistical functions

5.1. Maximum

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out

The intermediate value "O" is equal to the maximum value at the inputs In1...In8.
The module performs the operation only if all the configured inputs have a certain value.

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

5.2. Minimum

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out

The output value "Out" is equal to the minimum value at the inputs In1...In8.
The module performs the operation only if all the configured inputs have a certain value.

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

5.3. Average

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out

The intermediate value "O" is equal to the arithmetic mean of the sum of all values on the configured inputs. The module performs the operation only if all the configured inputs have a certain value.

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

5.4. Average without extremes

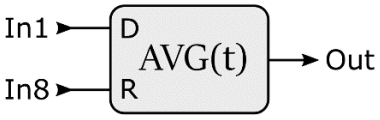
Inputs:
Data: {In1…In8}
Output:
O ⇒ Out

The intermediate value "O" is equal to the arithmetic mean of the sum of all the values on the configured inputs, excluding one largest and smallest value. When setting up the use of two inputs, the module works as a normal average. The module performs the operation only if all the configured inputs have a certain value.

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

5.5. Sliding average

Inputs:
Data: In1 (D)
Reset: In8 (R)
Output:
O ⇒ Out



Parameters:

«Window size»
Possible values: {2…**5**…20} probes

The number of cells in the internal buffer for storing data from In1 for averaging.

«Probe period»
Possible values: {1...**5**...3600} second

The time intervals after which the value will be saved from the In1 input to the next cell of the internal buffer.

«Result sending cycle»
Possible values: {1...**3**...20} probes

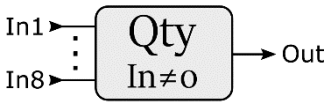
The number of measurements through which the module will calculate the average of all values of the internal buffer. The measurement result will be displayed every [Probe period] * [Result sending cycle] seconds.

The intermediate value "O" is equal to the arithmetic mean of the values stored in the internal buffer of the module.

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

5.6. Quantity of non-zero values

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out



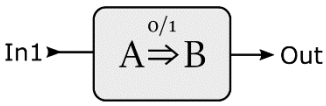
The intermediate value "O" is equal to the number of inputs that have a non-zero value.

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

6. Other functions

6.1. Logical conversion / Gate

Input:
In1
Output:
O ⇒ Out

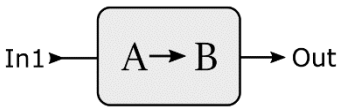


The intermediate value of "O" is 1 when the input value of In1 is not zero. Otherwise, the intermediate value of "O" is 0.

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

6.2. Format conversion / Gate

Input:
In1
Output:
O ⇒ Out

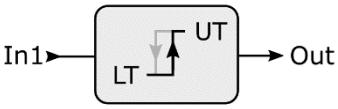


The intermediate value "O" is equal to the formatted input value (The format of the output variable is specified in the settings).

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

6.3. Switch with 2 thresholds

Input:
In1
Output:
O ⇒ Out



The intermediate value "O" is equal to the value 1 in the case when the input value is greater than the threshold UT.
The intermediate value "O" is equal to the value 0 in the case when the input value is less than the threshold LT.

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

6.4. Differential switch

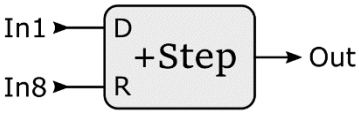
Inputs:
Data: In1 (D)
Reset: In8 (R)
Output:
O ⇒ Out

The intermediate value "O" is equal to the value 1 in the case when the difference between the average values of the current and previous measurement period has a positive value. The output value "O" is equal to the value 0 in the case when the difference between the average values of the current and previous measurement period has a negative value.

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

6.5. Pulse counter

Inputs:
Data: In1 (D)
Reset: In8 (R)
Output:
O ⇒ Out

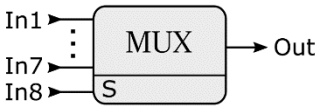


The intermediate value "O" increases by an amount equal to a step (set in the settings) for each non-zero value at the input starting from the specified initial value.

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

6.6. Multiplexer

Inputs:
Data: {In1…In7}
Select: In8 (S)
Output:
O ⇒ Out

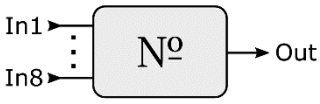


The intermediate value "O" is equal to the value at the selected input In1... In7. The input is selected by the In8(S) selection object

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

6.7. Input number indicator

Inputs:
Data: {In1…In8}
Output:
O ⇒ Out



The intermediate value "O" is equal to the input number In1...In8, on which a non-zero value was last recorded. For input In1, 0 is issued, for input In2, 1 is issued, and so on...

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

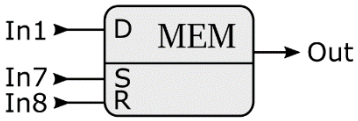
6.8. Staircase timer

Inputs:
In1
Output:
O ⇒ Out

Further steps with the value "O" are described in Chapter 7. Configuring functions with binary output

6.9. Memory cell

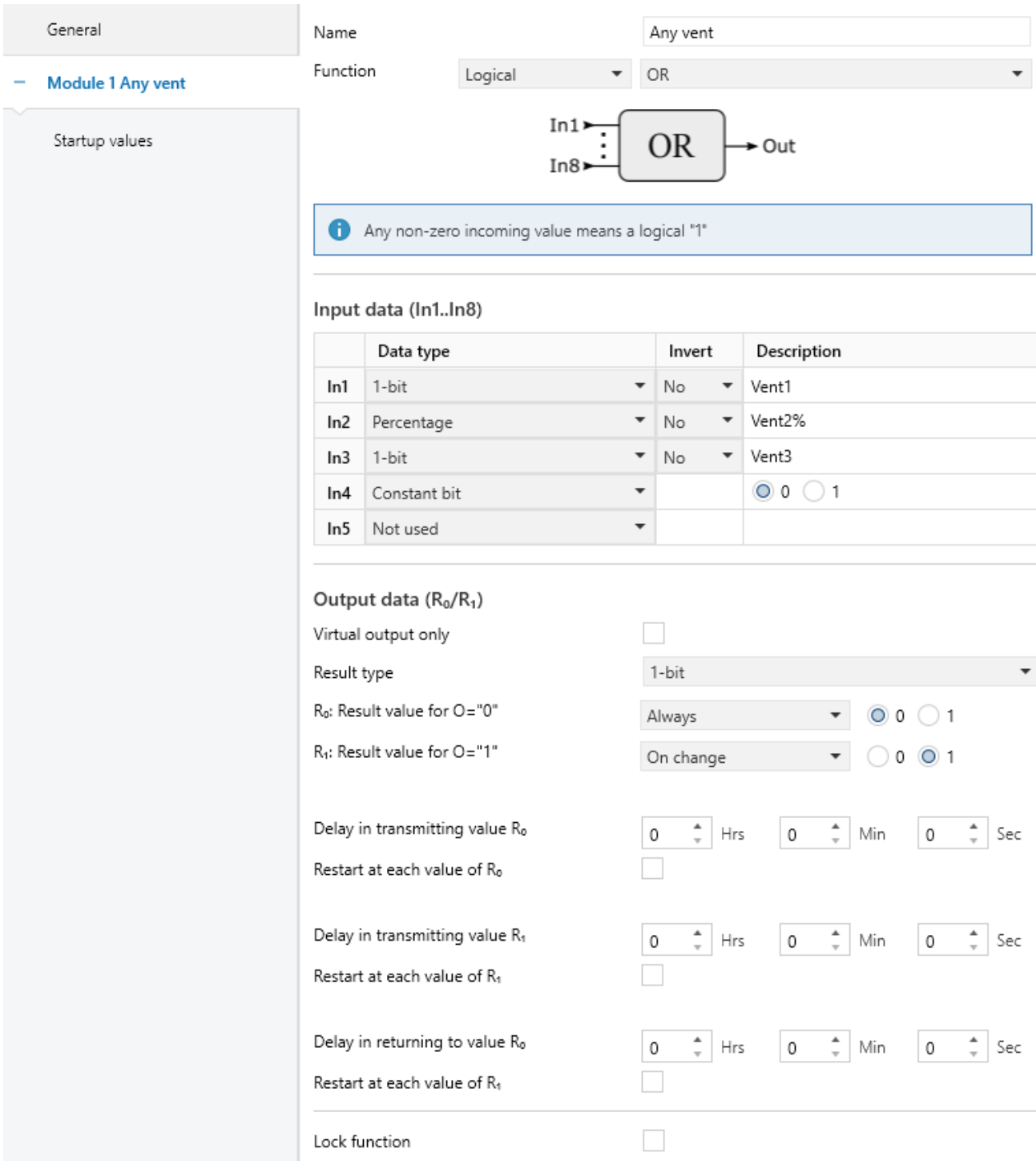
Inputs:
Data: In1 (D)
Store: In7 (S)
Read: In8 (R)
Output:
O ⇒ Out



When writing "1" to the input of the Store Object (In7), the module stores the value from the input In1 in memory. The module transmits the stored value to "O" each time it receives "1" at the input of the Read Object (In8).

Further steps with the value "O" are described in Chapter 8. Configuring functions with numeric output

7. Configuring functions with binary output



The result of executing a module with binary output can be one of two arbitrary constants R_0 and R_1 of one of the 10 data types defined in the "Result type" parameter. Optionally, the transmission of the result for one of the values can be disabled. You can also define a delay for transmitting each of the resulting constants, as well as for the constant r_1 the time after which the output will automatically return to the value r_0. The execution of any module can be blocked through an additional communication object "Lock". Any input parameter of any module with binary output can be a communication object of any type {"1-bit", "1-byte unsigned", "1-byte signed", "2-bytes unsigned", "2-bytes signed", "4-bytes signed", "2-bytes float", "Percentage"} or a bit constant {0,1} or the output of any other computing module ("Virtual module outputs")), while any value other than 0 is considered a logical "1".

«Only virtual output»
Possible values: {Off/On}.

When this setting is selected, the object corresponding to the Out of this module is hidden from the communication objects. In this case, the output of this module can be used as an input for other calculation modules

«Result type»
Possible values: {«1-bit», «1-byte unsigned», «1-byte signed», «2-bytes unsigned», «2-bytes signed», «4-bytes signed», «2-bytes float», «Scene», «Scene store», «Percentage»}

The parameter value determines the type of output communication object for this computing module. As a result, it is proposed to select two values of the selected type (R_0 and r_1 for intermediate values "0" and "1".

«R_0: Result value for O=”0”»

Setting the value of the selected data type to be transmitted to the Out output when the intermediate result of executing the binary calculation module is equal to "0".

Definition of transfer conditions:

- «Always»: value R_0 is passed to the Out output for each calculated intermediate result O=”0”, even if this value was also equal to “0” before that
- «On change»: value R_0 is passed to the Out output only if the value of the intermediate result O was equal to ”1” before this calculation, that is, it has changed
- «Disabled»: value transfer for the intermediate value O=”0” is disabled

«R_1: Result value for O=”1”»

Setting the value of the selected data type to be transmitted to the Out output when the intermediate result of executing the binary calculation module is equal to "1".

Definition of transfer conditions:

- «Always»: value R_1 is passed to the Out output for each calculated intermediate result O=”1", even if before that this value was also equal to “1”
- «On change»: value R_1 is passed to the Out output only if the value of the intermediate result O was equal to ”0” before this calculation, that is, it has changed
- «Disabled»: value transfer for the intermediate value O=”1” is disabled

«Delay in transmitting value R_0»
When: Condition value for the parameter "R_0: Result value for O="0" " is not equal to "Disabled"
Possible values: {0...18} hours, {0...59} minutes, {0...59} seconds

The time interval for which the sending of the R_0 value will be delayed.

«Restart at each value of R_0»
Possible values: {Off/On}

Restarting the timer for each calculated value of R_0

«Delay in transmitting value R_1»
When: Condition value for the parameter "R_1: Result value for O="1" " is not equal to "Disabled"
Possible values: {0...18} hours, {0...59} minutes, {0...59} seconds

The time interval for which the sending of the R_1 value will be delayed.

«Restart at each value of R_1»
Possible values: {Off/On}

Restarting the timer for each calculated value of R_1

«Delay in returning to value R_0»
When: Condition value for the parameter "R_0: Result value for O="0" " is not equal to "Disabled"

Possible values: {0...18} hours, {0...59} minutes, {0...59} seconds

The time interval after which the value of the result r_1 will be changed to R_0.

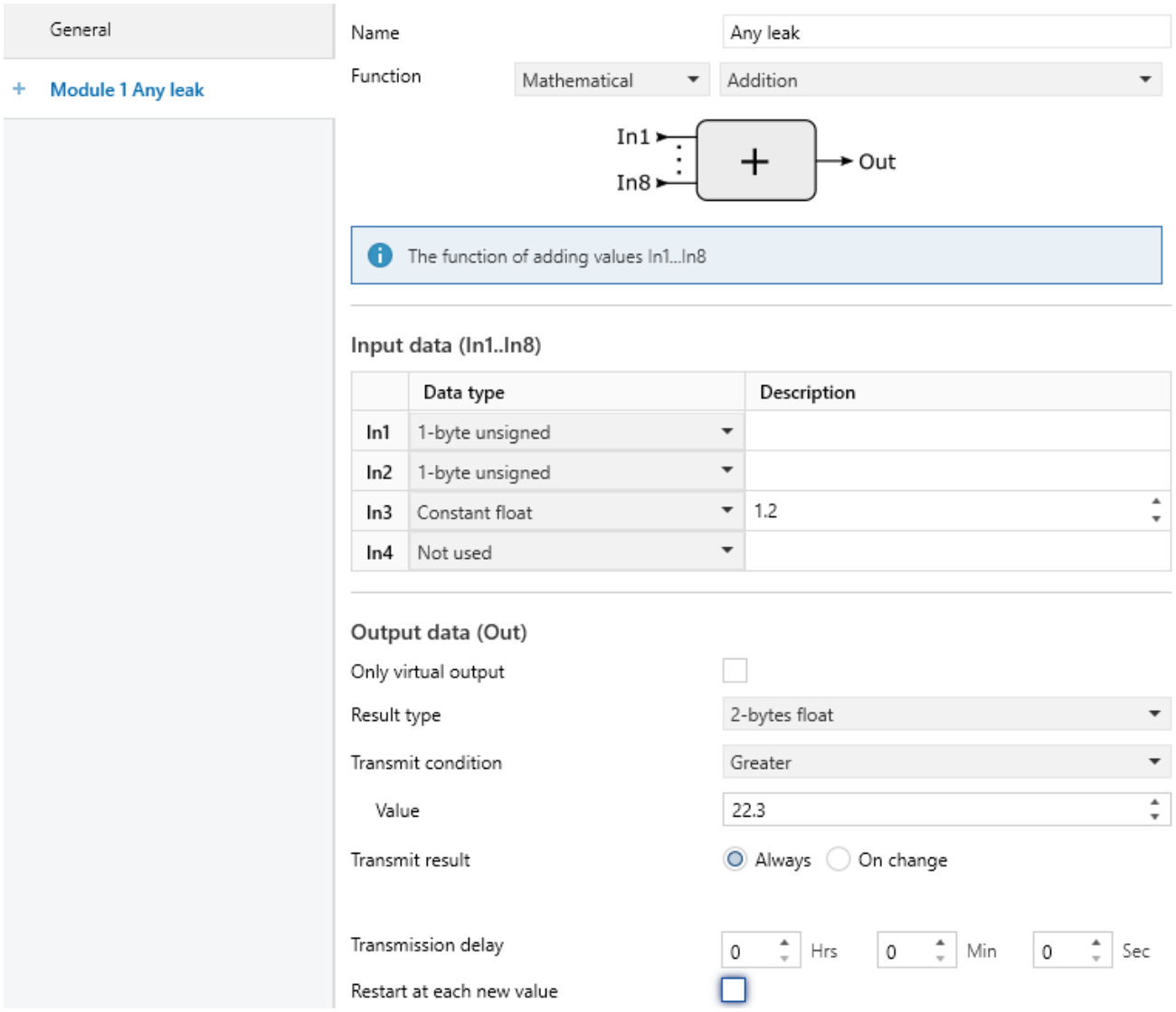
«Restart at each value of R_1»
Possible values: {Off/On}

Restarting the timer for each calculated value of R_1

«Lock function»
Possible values: {Off/On}

The configuration is described in Chapter 1.2. Blocking module execution

8. Configuring functions with numeric output



Any input parameter of any module with a numeric output can be a communication object of any type ("1-bit", "1-byte unsigned", "1-byte signed", "2-bytes unsigned", "2-bytes signed", "4-bytes signed", "2-bytes float", "Percentage", «Constant integer», «Constant float» or the output of any other computing module ("Virtual module outputs").

«Only virtual output»
Possible values: {Off/On}.

When this setting is selected, the object corresponding to the Out of this module is hidden from the communication objects. In this case, the output of this module can be used as an input for other calculation modules

«Result type»
Possible values: {«1-byte unsigned», «1-byte signed», «2-bytes unsigned», «2-bytes signed», «4-bytes signed», «2-bytes float», «Scene», «Scene store», «Percentage»}

Parameter value determines the type of output communication object for this computing module. If the value calculated by the module does not fit into the selected result type, then the value "1" is set to the communication object "Error"

«Transmit condition»
Possible values: {«Pass all», «Equal», «Not equal», «Greater», «Greater or equal», «Less», «Less or equal»}

Setting up the filtering condition for transmitting the calculated value to the output. If the value is "Pass all", all values are transmitted unconditionally, otherwise only if they are successfully compared with the set constant of the additional parameter "Value"

«Transmit result»
Possible values: {«Always», «On change»}

- «Always»: Calculated value is passed to the Out output for each result calculated by the module
- «On change»: Calculated value is passed to the Out output only if it has changed relative to the previously calculated one

«Transmission delay»
Possible values: {0...18} hours, {0...59} minutes, {0...59} seconds

The time interval for which the sending of the calculated value will be delayed.

«Restart at each new value»
Possible values: {Off/On}

Restarting the timer for each calculated value

«Lock function»
Possible values: {Off/On}

The configuration is described in Chapter 1.2. Blocking module execution

9. Communication objects

N = 11*[serial number of the module]

ID	Name/Function	I/O	Type	Flags
Conditions				
Description				
N	Module X In1 «Name». Input «Datatype»	I	Depends on settings	CWTU
Always				
N+1	Module X In2 «Name». Input «Datatype»	I	Depends on settings	CWTU
Always				
N+2	Module X In3 «Name». Input «Datatype»	I	Depends on settings	CWTU
Input In2 is activated				
N+3	Module X In4 «Name». Input «Datatype»	I	Depends on settings	CWTU
Input In3 is activated				
N+4	Module X In5 «Name». Input «Datatype»	I	Depends on settings	CWTU
Input In4 is activated				
N+5	Module X In6 «Name». Input «Datatype»	I	Depends on settings	CWTU
Input In5 is activated				
N+6	Module X In7 «Name». Input «Datatype»	I	Depends on settings	CWTU
Input In6 is activated				
N+7	Module X In7 «Name». Input «Datatype»	I	Depends on settings	CWTU
Input In7 is activated				
N+8	Module X In1 «Name». Output	I	Depends on settings	CRT
If the "Virtual output only" option is not activated				
N+9	Module X «Name». Lock	I	DPT_Enable	CW
If the "Lock function" setting is selected				
N+10	Module X «Name». Error	I	DPT_Alarm	CRT
Operations that may cause an error or overflow				