**CHOISE**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\2E32434F.tmp | Logical N/M Selector |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Choise.html#Introduction)  
[**Structure**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Choise.html#Structure)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Choise.html#Attributes)  
[**Binary automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

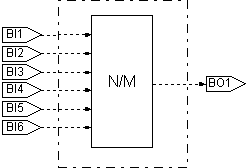
**Introduction**

The CHOISE module can be used as a logical n/m selector for binary signals. The maximum number of input signals is 6. The number m is always the same as the number of defined input signals. The number n is given by user.

The output signal is in 'True' state if at least n of the input signals are in 'True' state. Otherwise the output signal is in 'False' state. If the number n has been defined bigger than the number of input signals, the module behaves as normal AND gate.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Choise.html#Contents)

**Structure**

*Figure 1. Logic behind the CHOISE*

Consider a CHOISE module connected in an automation model. The created structure is shown in Figure 1 and the relevant parameters for this kind of a logical structure are listed in Table 1

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Relevant parameters for a CHOISE* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Value** | **Note** |
| CH\_N Number of TRUEs required to output TRUE | IN | 3 |  |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Choise.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 2. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| CH\_N Number of TRUEs required to output TRUE | IN | 2 | Integer number defining how many of the input signals at least should be in 'TRUE' state before the output signal turn to 'TRUE' state. |
| CH\_INPUT\_SIGN | ON(6) | 'OV' | Input signals (binary), maximum 6 signals can be defined, inputs that are not defined have no effect to the calculation. |
| CH\_OUTPUT\_SIGN | ON | \* | Output signal (binary) |

**ANALOG\_SIGNAL**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| **C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\55476AEB.tmp** | - |
| **C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\77136F31.tmp** | - |
| **C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\519B6967.tmp** | External |
| **C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\91AFF0D.tmp** | External I/O |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Analog_Signal.html#Introduction)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Analog_Signal.html#Attributes)  
[**Signal**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/ausign.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**Introduction**

ANALOG\_SIGNAL is used to connect two automation modules, i.e. an analog output terminal of an automation module with an analog input terminal of another automation module. ANALOG\_SIGNAL must never be connected into more than one module as an output signal. However, one signal can be an input signal for more than one module.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Analog_Signal.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 2. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| ANALOG\_VALUE | DO | 0.0 | Value of the analog signal |
| ANALOG\_VALUE\_OLD | DO | 0.0 | Value of the analog signal in the previous time step |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Analog_Signal.html#Contents)  
[**Signal**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/ausign.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**BINARY\_SIGNAL**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\D8365687.tmp | - |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\5199092D.tmp | - |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\EA44A0C3.tmp | External |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\D59AC9.tmp | External I/O |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Binary_Signal.html#Introduction)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Binary_Signal.html#Attributes)  
[**Signal**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/ausign.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**Introduction**

BINARY\_SIGNAL is used to connect two automation modules, i.e. a binary output terminal of an automation module with a binary input terminal of another automation module. BINARY\_SIGNAL must never be connected into more than one module as an output signal. However, one signal can be an input signal for more than one module.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Binary_Signal.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 2. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| BINARY\_VALUE | LO | F | Value of the binary signal |

**AND**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\3BB4D0BF.tmp | AND-3 (AND) |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\3843D825.tmp | AND-5 (AND) |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/AND.html#Introduction)  
[**Structure**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/AND.html#Structure)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/AND.html#Attributes)  
[**Binary automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

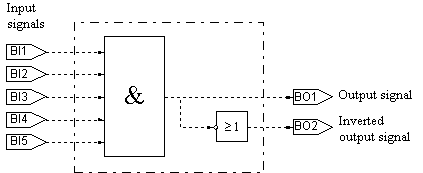
**Introduction**

The AND module can be used to perform a logical AND operation for binary signals. The maximum number of input signals is 30. Non defined input signals can be between the defined ones; they have no effect to the calculation. The output is 'True', if the inputs that are not inversed are in 'True' state, and if the inputs that are inversed are in 'False' state. In addition to the normal output signal, there is also an inversed output signal.

*Hint:* The graphical appearance does not change, when the user utilizes the inverse of an input. To avoid this disinformation in the graphical document, one can draw a small circle close to the inversed input terminal.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/AND.html#Contents)

**Structure**

*Figure 1. Logic behind the AND*

Consider an AND module connected in an automation model. The created structure is shown in Figure 1**.**

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/AND.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| AND\_INPUT\_SIGN | ON(30) | 'OV' | Names of the input signals |
| AND\_INPUT\_AS\_INV Is input inversed? | LO(30) | F | Is the corresponding input inversed? For example, if the 3. value of the and\_input\_as\_inv attribute array is set to 'True', the 3. input must be 'False' in order to allow the output get 'true'. |
| AND\_OUTPUT\_SIGN | ON | \* | Output signal |
| AND\_OUTINV\_SIGN | ON | \* | Inverse output signal |
| AND\_OUTPUT\_VALUE Output value | LO | F | Output value |

**NOT**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| **C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\AC6C71A3.tmp** | NOT |

**Contents**

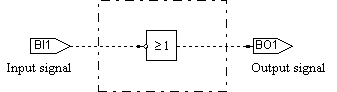
[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/NOT.html#Introduction)  
[**Structure**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/NOT.html#Structure)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/NOT.html#Attributes)  
[**Binary automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**Introduction**

NOT is a logical module that can be used to invert the state of the binary signal. This module type has only one input and output signal. The state of the output signal is always inverse of the input signal.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/NOT.html#Contents)

**Structure**

*****Figure 1. Logic behind the NOT module*

Consider a NOT module connected in an automation model. The created structure is shown in Figure 1**.**

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/NOT.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| NOT\_INPUT\_SIGN | ON\* | 'OV' | Name of the input signal (binary) |
| NOT\_OUTPUT\_SIGN | ON | \* | Name of the output signal (binary) |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/NOT.html#Contents)  
[**Binary automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**OR**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| **C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\53E3CEFA.tmp** | OR-3 (OR) |
| **C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\FC8EE118.tmp** | OR-5 (OR) |

**Contents**

[**Introduction**](http://127.0.0.1:49241/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/OR.html#Introduction)  
[**Structure**](http://127.0.0.1:49241/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/OR.html#Structure)  
[**Attributes**](http://127.0.0.1:49241/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/OR.html#Attributes)  
[**Binary automation**](http://127.0.0.1:49241/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:49241/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

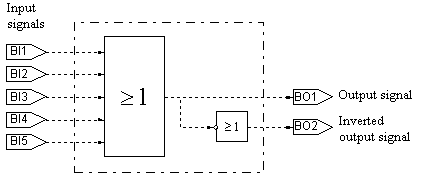
**Introduction**

The OR module is used to perform the logical operation 'OR' for binary signals. Maximum number of input signals is 30. It is not necessary to define all inputs, the non defined inputs have no effect to the logical operation.

The output signal of the module is in 'True' state if anyone of the defined inputs are in 'True' state. Output is 'False' only if all the defined inputs are in 'False' state.

[**To contents of this component**](http://127.0.0.1:49241/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/OR.html#Contents)

**Structure**

****

*Figure 1. Logic behind the OR*

Consider an OR module connected in an automation model. The created structure is shown in Figure 1**.**

[**To contents of this component**](http://127.0.0.1:49241/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/OR.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| OR\_INPUT\_SIGN | ON\*(30) | 'OV' | Names of the input signals. Maximum 30 input signals can be defined. |
| OR\_OUTPUT\_SIGN | ON | \* | Name of the output signal |
| OR\_OUTINV\_SIGN | ON | \* | Name of the inversed output signal |
| OR\_OUTPUT\_VALUE Output value | LO | F | Output value of the module |

**FLIP\_FLOP**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\538E9D41.tmp | Flip-flop |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Flip_flop.html#Introduction)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Flip_flop.html#Attributes)  
[**Binary automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**Introduction**

The FLIP\_FLOP (bistable) is a logical module, that can be used as a memory in logic circuits. The module has two input signals, set and reset, and two output signals, output and its inverse. The output indicates, which one of the two input signals was most recently in 'True' state. The value of the inverse signal is always the opposite of the output signal. The output of the module depends on the values of the input signals as follows:

* if both input signals are in 'False' state, the value of output does not change (the output value can be changed with the attribute FF\_OUTPUT\_VALUE),
* if SET input is in 'True' state, the output will be in 'True' state,
* if only RESET input is in 'True' state, the output will be in 'False' state,
* if both inputs are in 'True' state and reset is not dominating, the output will be 'True',
* if both inputs are in 'True' state and reset is dominating, the output will be 'False'.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Flip_flop.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| FF\_OUTPUT\_VALUE Output value | LO | F | The output value of the flip-flop (the initial output value is defined with this attribute if both set and reset signals have the value 'false') |
| FF\_RESET\_SIGN | ON | 'OV' | Name of the reset signal |
| FF\_SET\_SIGN | ON | 'OV' | Name of the set signal |
| FF\_RESET\_DOMINATES Is reset dominating? | LO | F | Is reset dominating? If 'true', the output gets 'false' in the case of having both inputs in 'true' state. |
| FF\_OUTPUT\_SIGN | ON | \* | Name of the output signal |
| FF\_OUTINV\_SIGN | ON | \* | Name of the inverse output signal |

**MEASUREMENT**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\A6FA91F7.tmp | General (Measurement) |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Introduction)  
[**Structure**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Structure)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Attributes)  
[**Special**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Special)  
[**Normal operation mode**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Normal_operation_mode)  
[**Simulation mode**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Simulation_mode)  
[**Faults and malfunctions**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Faults_and_malfunctions)  
[**Error messages**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Error_messages)  
[**Measurement**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aumea.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

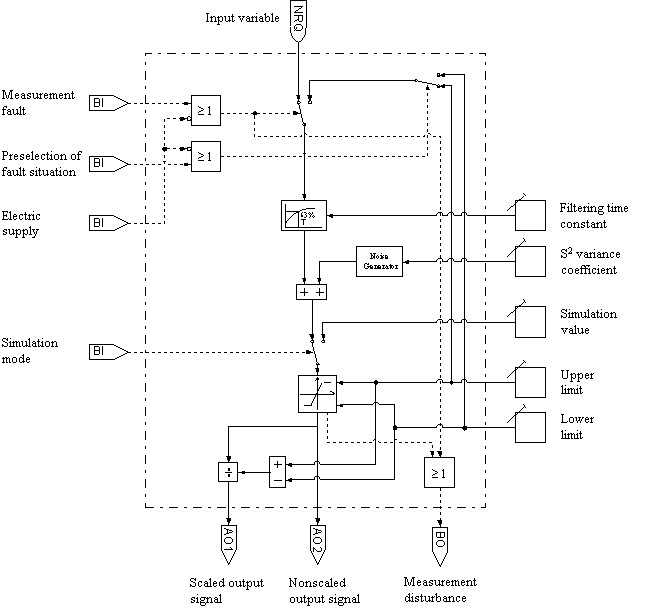
**Introduction**

The MEASUREMENT module can be used as an general interface from process variables to control circuits. The module has two analog output signals: a scaled signal (value ranging from 0 ... 1) and a nonscaled one.

The measured variable is defined by object quartet (module name + attribute + j-index + i-index). Typically, only object pair (module name + attribute) is needed. It is possible to connect the measurement to any double precision, real, integer or logical variable, which can be identified by object quartet.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Contents)

**Structure**

*Figure 1. Logic behind the MEASUREMENT*

Consider a MEASUREMENT module connected in an automation model. The created structure is shown in Figure 1 and the relevant parameters in a simple case for this kind of a logical structure are shown in Table 1.

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Relevant parameters for a simple MEASUREMENT* | | | |
| **Attribute Property name** | **Type** | **Value** | **Note** |
| ME\_COEFFICIENT Scale coefficient | DO | 1.0 |  |
| ME\_BIAS Scale bias | DO | 0.2 |  |
| ME\_LOW\_LIMIT Low limit | DO | 1E-9 |  |
| ME\_HIGH\_LIMIT High limit | DO | 100.0 |  |
| ME\_TIME\_CONST Time constant | DO | 1.0 |  |
| ME\_VARIANCE Variance of measurement noise | DO | 0.1 |  |

It is also important to choose the desired referenced attribute from ME\_PROCESS\_COMPONENT.  
[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 2. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| ME\_PROCESS\_COMPONENT | OQ | NULL NULL 1 1 | Object quartet defining the process component and attribute and possible indexes into which the measurement is connected |
| ME\_COEFFICIENT Scale coefficient | DO | 1.0 | Scaling coefficient with which the measured value is multiplied. |
| ME\_BIAS Scale bias | DO | 0.0 | Bias term. The measured value is multiplied with the coefficient before adding up the bias. |
| ME\_LOW\_LIMIT Low limit | DO | 0.0 | Lower limit of the measurement output range |
| ME\_HIGH\_LIMIT High limit | DO | 100.0 | Upper limit of the measurement output range |
| ME\_VALUE State matrix | DO | (2,2) | Matrix indicating inside values of the component: (1,1) & (1,2) = current time step's raw measurement (unlimited) (2,1) & (2,2) = current time step's filtered measurement (unlimited) |
| ME\_TIME\_CONST Time constant | DO | 1.0 | Filtering time constant of the measurement (in seconds). Not used in scanner or sampling modes (see ME\_MEASUREMENT\_MODE). |
| ME\_VARIANCE Variance of measurement noise | DO | 0.0 | Variance of the measurement noise |
| ME\_FAULT\_ON Fault on | LO | F | Logical input by which it is possible to simulate the fault of the measurement |
| ME\_FAULT\_VALUE Value of output at fault | DO | 0.0 | Value of the non-scaled output at electrical (ME\_ELECTRIC\_SUPPLY = F) and **other faults (ME\_FAULT\_ON = T)**. |
| ME\_SIM\_MODE\_ON Simulation mode on | LO | F | Logical input, when this input is TRUE, output of the measurement follows the value of the simulation input with a ramp defined with ME\_SIM\_VAL\_RAMP\_TIME. It should be noted that the value of ME\_MALFUNCTION must be -1 or 5 in order for the output to follow the simulation value. |
| ME\_SIMULATION\_VALUE Simulation value | DO | 0.0 | Simulation input of the measurement |
| ME\_ELECTRIC\_SUPPLY Is electricity supplied | LO | T | Logical input, indicates if measurement has electricity or not |
| ME\_MALFUNCTION Measurement malfunction | IN | -1 | Definition of normal or abnormal function of the general measurement. A value from -1 to 5 is given depending on the desired function: -1 = normal function 0 = normal function, but variance does not give noise. Simulation mode (ME\_SIM\_MODE ON = T) has no effect. 1 = measurement is stuck. Simulation mode (ME\_SIM\_MODE ON = T) has no effect. 2 = to upper limit. Simulation mode (ME\_SIM\_MODE ON = T) has no effect. 3 = to lower limit. Simulation mode (ME\_SIM\_MODE ON = T) has no effect. 4= noisy (variance must be given). Simulation mode (ME\_SIM\_MODE ON = T) has no effect. 5 = simulation |
| ME\_MEASUREMENT\_MODE Measurement mode | IN | 0 | Measurement mode. 0 = normal mode: defined variable is measured and filtered 1 = scanner 2 = unfiltered sampling |
| ME\_SCANNING\_TIME Time needed for one scan | DO | 30 | Time for scanning (me\_measurement\_mode = 1) or period between sampling (me\_measurement\_mode = 2). |
| ME\_SCANNER\_STATE | IN | 0 | State needed in calculation of scanning and sampling modes (me\_measurement\_mode = 1/2). |
| ME\_SCANNER\_CLOCK | DO | 0 | Clock needed in calculation of scanning and sampling modes (me\_measurement\_mode = 1/2). |
| ME\_INTEGRAL | DO | 0 | Cumulative sum needed in calculation of scanning mode (me\_measurement\_mode = 1). |
| ME\_SCANNER\_RESTING\_TIME | DO(2) | 0 | Resting time, time when scanning is not proceeding. Own time for the odd and the even scan. Used in scanning mode (me\_measurement\_mode = 1). |
| ME\_FAULT\_ON\_S | ON\* | NULL | Signal input of the attribute 'ME\_FAULT\_ON' |
| ME\_SIM\_MODE\_ON\_S | ON\* | NULL | Signal input of the attribute 'ME\_SIM\_MODE\_ON' |
| ME\_SIMULATION\_S | ON\* | NULL | Signal input of the attribute 'ME\_SIMULATION\_VALUE' |
| ME\_ELECTRIC\_SUPPLY\_S | ON\* | NULL | Signal input of the attribute 'ME\_ELECTRIC\_SUPPLY' |
| ME\_SCALED\_OUT\_SIGN | ON | \* | Name of the scaled output signal of the module |
| ME\_NONSCALED\_OUT\_SIGN | ON | \* | Name of the nonscaled output signal of the module |
| ME\_DISTURB\_IND\_SIGN | ON | \* | Name of the alarm signal. Signal is TRUE if electric supply of the module fails, if the fault situation has been selected or if the measured variable is out of defined measurement range |
| ME\_NEW New component | LO | T/F | Logical attribute, should not be modified by the user |
| ME\_OUTPUT\_VALUE Output value | DO | 0.0 | Output value of the module |
| ME\_SCALED\_OUTPUT\_VALUE Scaled output value | DO | 0.0 | Scaled output value of the module (see low and high limits) |
| ME\_EXCEED\_LIMITS Limit exceeding option | IN | 0 | Integer flag to tell whether exceeding low and high limits is allowed. The scaled output may in cases of allowed exceeding be <0.0 or >1.0! For more information, look table 3. |
| ME\_SIM\_VAL\_RAMP\_TIME Simulation value ramp time | DO | 0.0 | Ramp time (s) for simulation value. When measurement module is changed to simulation mode the output will ramp from the current value to the simulation value in the time defined with this attribute. If the simulation value changes when in simulation mode the new simulation value is reached again in this time (not faster, not slower). When simulation mode is switched off, the output returns to the process value in one time step. |
| ME\_SEED Seed number | IN | (system time based) | The manually given 1st seed number. Default value is generated based on system time. User can manually change it. Utilised if CURRENT\_SEED = 0. |
| ME\_CURRENT\_SEED Current seed | DO | 0 | Current seed number used to calculate next noise value. If 0 will be generated based on the value given in the SEED attribute. |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Contents)

|  |  |  |
| --- | --- | --- |
| *Table 3. Limit exceeding option* | | |
| **Value** | **Low limit exceedeng allowed?** | **High limit exceeding allowed?** |
| 0 | no | no |
| 1 | yes | yes |
| 2 | no | yes |
| 3 | yes | no |

**Special**

MEASUREMENT module has the following operation modes:

* normal operation mode (three submodes: filtered measurement, scanner, sampling)
* simulation mode,
* faults and malfunctions

See the [**Automation modelling guide**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/guides/autgui/autgui.htm#Using_measurement) for instructions on how to connect a measurement.

**Normal operation mode**

**Filtered measurement**

In normal operation mode the output signals follow the changes in the measured variable. The new measurement is filtered in the following way:

Mf(k) = Tf /(dt+Tf) \* Mf(k-1) + dt /(dt+Tf) \* M(k)

Mf(k)      Filtered measurement value at the current time step  
Mf(k-1)   Filtered measurement value from the previous time step  
M(k)       Measured value at the current time step  
dt           Time step  
Tf           Filtering time constant

If a non-positive filtering time constant is given, 0.0001 is used instead. The filtered measurement value is multiplied with the coefficient (ME\_COEFFICIENT), and the bias (ME\_BIAS) is added up. After that, the measurement noise, defined by giving a nonzero value for the variance, is taken into account:

M(k) = coefficient \* Mf(k) + bias + noise

If the measurement value M(k) is out of the measurement range, the output signals are limited either to the lower or to the upper limit and the DISTURB\_IND\_SIGN gets value 'True'. The scaled output signal is scaled according to the given measurement range.

**Scanner**

In the scanner submode (ME\_MEASUREMENT\_MODE= 1), during the time me\_scanning\_time, the defined variable is measured every time step (unfiltered) and sum of the measured value times the passed simulation time (me x dt) is cumulated. In the end of the time, the cumulative sum is divided by the me\_scanning\_time and sent as output. Before new scanning period, the user defined me\_scanner\_resting\_time is waited. Two different times can be defined.

**Sampling**

In the sampling submode (me\_measurement\_mode = 2), an unfiltered sample of the measured variable is taken each time the ME\_SCANNING\_TIME has passed.

**Simulation mode**

In simulation mode, the output signals follow the simulation value input, and filtering, coefficient&bias, noise and faults are passed. The simulation value can either be constant (ME\_SIMULATION\_VALUE) or depend on an analog signal (ME\_SIMULATION\_S). It should be noted that the value of ME\_MALFUNCTION must be -1 or 5 in order for the output to follow the simulation value.

**Faults and malfunctions**

In fault mode, the output values move either towards the upper or lower limit of the measurement. The selection depends on the value of the fault preselection input. If the fault preselection input is 'True' the outputs move towards the lower limit.

In the case of electric supply fault, the output signals will move towards the lower limit of the measurement range.

Besides special operation by fault and simulation mode and electric supply attributes, it is possible to use attribute ME\_MALFUNCTION. A value from -1 to 5 is given depending on the desired function:

-1  normal function, other ways to define faults (e.g. me\_fault\_on) or simulation mode can be used  
0      normal function, except that giving variance does not give noise until the malfunction 4 is switced on, simulation mode has no effect   
1       measurement gets stuck (keeps its current value), simulation mode has no effect   
2       measurement goes to the upper limit, simulation mode has no effect   
3       measurement goes to the lower limit, simulation mode has no effect   
4       measurement is noisy (a positive variance must be given), simulation mode has no effect   
5       simulation, outputs of the measurement follow the simulation value

From the malfunctions 1-5 it follows, that the ME\_DISTURB\_IND\_SIGN gets value 'True'.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Contents)

**Error messages**

**4133 Check the measurement range of ...:** The upper limit of the measurement is smaller than the lower limit.

**4204 Measurement not properly connected:** A proper connection to measured item is missing. If both connections of the difference measurement are missing, two warnings are given.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Measurement.html#Contents)  
[**Measurement**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aumea.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**LV\_CHECKER**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\20FC5B33.tmp | Limit Value Checker |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Lv_checker.html#Introduction)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Lv_checker.html#Attributes)  
[**Binary automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**Introduction**

The LV\_CHECKER stands for limit value checker. It can simulate three different types of limit value checkers:

Type 0: "traditional Apros limit value checker"

Type 1: "high limit checker"

Type 2: "low limit checker"

The input value is compared to the limit value, considering possible hysteresis, in the following way:

If Type = 0 then

* If input > (limit value + hysteresis/2), then output 1 is true and output 2 is false
* If input < (limit value - hysteresis/2), then output 1 is false and output 2 is true.
* If (limit value - hysteresis/2) ≤ input ≤ (limit value + hysteresis/2), then the previous value of the output is held. When a new module is created, this 'previous value' is FALSE.

If Type = 1 then

* If input > (limit value), then output 1 is true and output 2 is false
* If input < (limit value - hysteresis), then output 1 is false and output 2 is true.
* If (limit value - hysteresis) ≤ input ≤ (limit value), then the previous value of the output is held. When a new module is created, this 'previous value' is FALSE.

If Type = 2 then

* If input > (limit value + hysteresis), then output 1 is true and output 2 is false
* If input < (limit value), then output 1 is false and output 2 is true.
* If (limit value) ≤ input ≤ (limit value + hysteresis), then the previous value of the output is held. When a new module is created, this 'previous value' is FALSE.

In normal modes (i.e. not forced modes, see below), output 2 is always the negation of output 1.

In forced output modes the output is determined differently. The forced mode is controlled with the attribute LVC\_OUTPUT\_OPTION, the possible modes are listed in Table 2.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Lv_checker.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| LVC\_LIMIT\_VALUE | DO | 0.0 | Limit value |
| LVC\_HYSTERESIS | DO | 0.0 | Hysteresis value |
| LVC\_OUTPUT1\_VALUE | LO | - | Output value |
| LVC\_LIMIT\_VALUE\_S | ON\* | NULL | Signal input for the attribute "lvc\_limit\_value" |
| LVC\_INPUT\_SIGN | ON | 'OV' | Name of the input signal (analog) |
| LVC\_OUTPUT\_SIGN\_1 | ON | \* | Name of the output signal 1 (binary) |
| LVC\_OUTPUT\_SIGN\_2 | ON | \* | Name of the output signal 2 (binary) |
| LVC\_NEW | LO | true | Logical value, should not be modified by the user |
| LVC\_OUTPUT\_OPTION | IN | 0 | Output mode, see table 2. |
| LVC\_TYPE | IN | 0 | 0=Traditional Apros Limit value checker, 1=High limit checker, 2=Low limit checker. Please see description above for more details |
| LVC\_IN\_CYCLE | IN | 0 | Indicator bit of whether the module belongs to a CYCLE (=1) or not (=0), set automatically. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 2. Output modes of LV\_CHECKER* | | | |
| **Value** | **Output 1** | **Output 2** | **Description** |
| 0 | T/F | F/T | normal mode, as described above |
| 1 | F | F | wire break, forced F-F mode |
| 2 | T | T | short circuit, forced T-T mode |
| 3 | T | F | forced T-F mode |
| 4 | F | T | forced F-T mode |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Lv_checker.html#Contents)  
[**Binary automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**GRADIENT**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\FF9FEEB9.tmp | Gradient |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Gradient.html#Introduction)  
[**Structure**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Gradient.html#Structure)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Gradient.html#Attributes)  
[**Analog automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/auana.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**Introduction**

The GRADIENT is a dynamic module, that can be used to limit the rate of change of a signal. The rate is limited separately for increase and for decrease direction:

* For increase, the maximum gradient used is |grad\_value\_upwards|
* For decrease, the maximum gradient used is -|grad\_value\_downwards|.

Three binary inputs (grad\_stop, grad\_fast and grad\_tracking\_on) controls the operation modes of the module. The GRADIENT has two basic operation modes:

* normal operation
* tracking.

In the normal operation mode (grad\_tracking\_on=F) the input signal (grad\_input\_sign) is followed. In the tracking mode, the tracking input signal (grad\_tracking\_sign) is followed. The output signal follows the changes of input signal with the allowed maximum gradient values. If the change in the input signal is slower than the defined maximum gradient, the value of the ouput is exactly the same as the input.

There are two options (sub modes) available in both modes normal and tracking:

* stop
* fast.

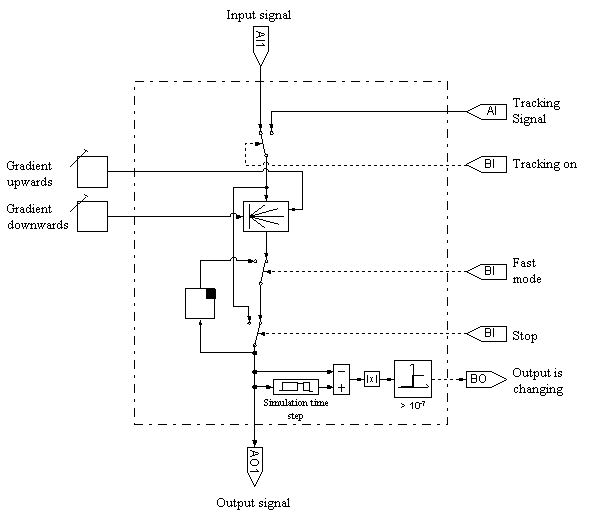
In the STOP mode (grad\_stop=T), the output signal is frozen, i.e. it will not change yet the input signal is changing.  
In the FAST mode (grad\_fast=T), the output of the module is following the input without any gradient limitation.  
The priority of these sub modes is STOP > FAST.

The attribute grad\_fast\_tracking offers an option that tracking is always performed without gradient limitation (despite of the value of the grad\_fast attribute).

The binary output signal 'grad\_outp\_changing\_sign' indicates (with value 'True'), if the output is changing.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Gradient.html#Contents)

**Structure**

*Figure 1. Logic behind the GRADIENT*

Consider a GRADIENT module connected in an automation model. The created structure is shown in Figure 1 and the relevant parameters for this kind of a logical structure are shown in Table 1.

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Relevant parameters for a GRADIENT* | | | |
| **Attribute Property name** | **Type** | **Value** | **Note** |
| GRAD\_VALUE\_UPWARDS Gradient value for increase | DO | 10.0 |  |
| GRAD\_VALUE\_DOWNWARDS Gradient value for decrease | DO | 10.0 |  |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Gradient.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 2. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| GRAD\_VALUE\_UPWARDS Gradient value for increase | DO | 10.0 | Maximum gradient upwards |
| GRAD\_VALUE\_DOWNWARDS Gradient value for decrease | DO | ­10.0 | Maximum gradient downwards (abs.value) |
| GRAD\_STOP Output freezed? | LO | F | Stop mode on? |
| GRAD\_FAST Gradients out of operation? | LO | F | Fast mode on? |
| GRAD\_TRACKING\_ON Tracking on? | LO | F | Tracking mode on? |
| GRAD\_FAST\_TRACKING | LO | F | Fast operation in tracking mode? |
| GRAD\_AGRRA1 Current and previous inputs | DO(1,2) |  | Current and previous inputs[i] |
| GRAD\_AGRRA2 Current and previous outputs | DO(1,2) |  | Current and previous outputs[i] |
| GRAD\_VALUE\_UPWARDS\_S | ON\* | NULL | Max. gradient upwards signal |
| GRAD\_VALUE\_DOWNWARDS\_S | ON\* | NULL | Max. gradient downwards signal |
| GRAD\_STOP\_S | ON\* | NULL | Stop mode on signal |
| GRAD\_FAST\_S | ON\* | NULL | Fast mode on signal |
| GRAD\_TRACKING\_ON\_S | ON\* | NULL | Tracking mode on signal |
| GRAD\_INPUT\_SIGN | ON | 'OV' | Input signal |
| GRAD\_TRACKING\_SIGN | ON | NULL | Tracking input signal |
| GRAD\_OUTPUT\_SIGN | ON | \* | Output signal |
| GRAD\_OUTP\_CHANGING\_SIGN | ON | \* | Output is changing signal |
| GRAD\_NEW New gradient module | LO | T/F |  |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Gradient.html#Contents)  
[**Analog automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/auana.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**BINARY\_DELAY**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\BB9C3F9F.tmp | Binary Delay |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Binary_delay.html#Introduction)  
[**Structure**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Binary_delay.html#Structure)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Binary_delay.html#Attributes)  
[**Binary automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

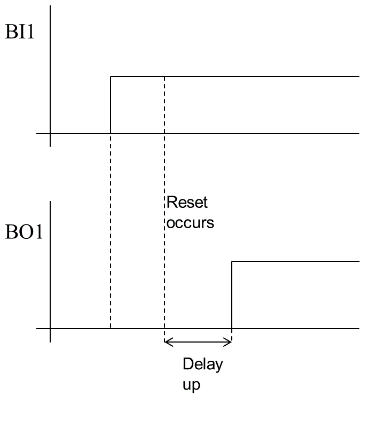
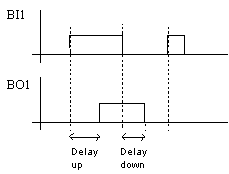
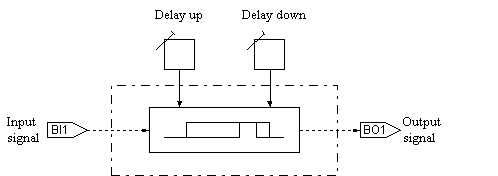
**Introduction**

The BINARY\_DELAY module can be used to delay binary signals. The state of the output signal will follow the state of the input signal after a defined delay time. The delay can be defined separately for changes from 'False' to 'True' (up) and from 'True' to 'False' (down). Both time delays can be constant or they can depend on some analog signals. In the case the duration of the input pulse is shorter than the defined delay time, no change will happen in the output signal.

User can reset the binary delay's remaining time usign DELAY\_RESET\_REMAINING\_TI. This means that if up or down delay is active, then its length will be extended by DELAY\_UP or DELAY\_DOWN.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Binary_delay.html#Contents)

**Structure**



*Figure 1. Logic behind the BINARY\_DELAY and step responses (note the output when the input pulse is shorter than the delay). On far right, the effect of resetting of remaining time.*

Consider a BINARY\_DELAY module connected in an automation model. The created structure is shown in Figure 1 and the relevant parameters for this kind of a logical structure are shown in Table 1.

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Relevant parameters for a BINARY\_DELAY* | | | |
| **Attribute Property name** | **Type** | **Value** | **Note** |
| DELAY\_UP Time delay for change: FALSE -> TRUE | DO | 1 | Delay in seconds |
| DELAY\_DOWN Time delay for change: TRUE -> FALSE | DO | 1 | Delay in seconds |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Binary_delay.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 2. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| DELAY\_UP Time delay for change: FALSE -> TRUE | DO | 1.0 | the time delay in seconds for the changes from 'FALSE' to 'TRUE', |
| DELAY\_DOWN Time delay for change: TRUE -> FALSE | DO | 1.0 | the time delay in seconds for the changes from 'TRUE' to 'FALSE', |
| DELAY\_UP\_S | ON\* | NULL | signal input for attribute 'delay\_up', |
| DELAY\_DOWN\_S | ON\* | NULL | signal input for attribute 'delay\_down', |
| DELAY\_INPUT\_SIGN | ON | 'OV' | input signal (binary) |
| DELAY\_OUTPUT\_SIGN | ON | \* | output signal (binary) |
| DELAY\_REMAINING\_TIME | DO | 0.0 | The remaining time for the next change of output value. |
| DELAY\_RESET\_REMAINING\_TI | LO | FALSE | The remaining time for the next change of output value is set back to DELAY\_UP or DELAY\_DOWN, depending which delay is active. This attribute will have effect at the next simulation time step. After the remaining time is set back to appropriate delay, this attribute is reset automatically to FALSE. |
| DELAY\_ACTIVE\_DELAY | IN | 0 | Shows which delay is active: 0 = none, 1 = up, 2 = down. |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Binary_delay.html#Contents)  
[**Binary automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**ANALOG\_SWITCH**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\39AFDCD7.tmp | Analog Switch |

**Contents**

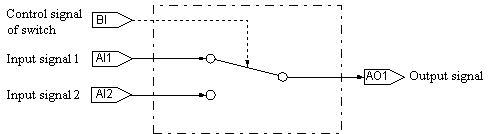
[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Analog_switch.html#Introduction)  
[**Structure**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Analog_switch.html#Structure)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Analog_switch.html#Attributes)  
[**Analog automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/auana.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**Introduction**

ANALOG\_SWITCH can be used as a selector for two analog signals. The value of the module output signal will follow the value of either of the two input signals. If the control input of the module is 'F', output signals follows the input signal 1 and if the control input is 'T' the output signal follows the input signal 2.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Analog_switch.html#Contents)

**Structure**



*Figure 1. Logic behind the ANALOG\_SWITCH*

Consider an ANALOG\_SWITCH module connected in an automation model. The created structure is shown in Figure 1.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Analog_switch.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 2. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| SWITCH\_CONTROL Switch position | LO | F | logic input, when TRUE, output signal follows input signal 2 |
| SWITCH\_CONTROL\_S | ON\* | NULL | signal input for attribute 'switch\_control' |
| SWITCH\_INP\_SIGN\_1 | ON | 'OV' | input signal (analog) |
| SWITCH\_INP\_SIGN\_2 | ON | 'OV' | input signal (analog) |
| SWITCH\_OUTP\_SIGN | ON | \* | output signal (analog) |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Analog_switch.html#Contents)  
[**Analog automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/auana.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**SET\_POINT**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\141BE413.tmp | Setpoint |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\E5A1CA99.tmp | Setpoint S (Setpoint) |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Set_point.html#Introduction)  
[**Structure**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Set_point.html#Structure)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Set_point.html#Attributes)  
[**User interface components**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/auuse.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**Introduction**

The SET\_POINT module can be used as set point generator in control circuits. The set point module has operation modes:

* normal operation
* tracking operation

The operation is defined by binary input SP\_TRACKING\_ON or the corresponding input signal. In normal operation the output follows the attribute SP\_VALUE. In tracking mode the output follows the value of the input signal and the SP\_VALUE attribute is updated accordingly.

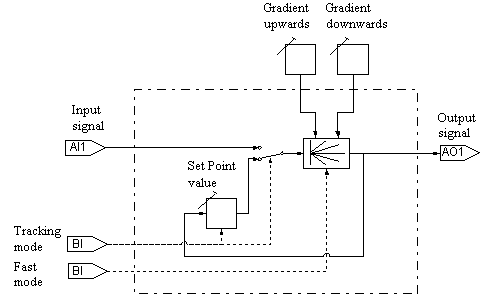
Furthermore, the output follows the given set value either:

* immediately or with a given gradient
* unlimited or limited with given minimum and maximum values

The operation is defined by binary inputs: SP\_FAST\_MODE\_ON and SP\_MINMAX\_ON. In fast mode (SP\_FAST\_MODE\_ON in 'True' state) the output of the module follows either the SP\_VALUE attribute or the input signal (if also SP\_TRACKING\_ON is in 'True' state) without any gradient limitation. In case of using minimum and maximum limits (SP\_MINMAX\_ON in 'True' state), the output never exceeds the given limits.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Set_point.html#Contents)

**Structure**



*Figure 1. Logic behind the SET\_POINT*

Consider a SET\_POINT module connected in an automation model. The created structure is shown in Figure 1 and the relevant parameters for this kind of a logical structure are shown in Table 1.

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Relevant parameters for a SET\_POINT* | | | |
| **Attribute Property name** | **Type** | **Value** | **Note** |
| SP\_VALUE Setpoint value | DO | 5 |  |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Set_point.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 2. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| SP\_VALUE Setpoint value | DO | 0.0 | Manually given set point value, or the value of the input signal in tracking mode |
| SP\_GRADIENT\_UP Gradient upwards | DO | 60.0 | Change rate (1/min) of the output signal, when set point is increasing |
| SP\_GRADIENT\_DOWN Gradient downwards | DO | ­60.0 | Change rate (1/min) of the output signal, when set point is decreasing |
| SP\_TRACKING\_ON Tracking on? | LO | F | Logical input, which turns the set point module to tracking mode |
| SP\_FAST\_MODE\_ON Fast mode on? | LO | F | Logical input, which turns the set point module to fast mode |
| SP\_GRADIENT\_UP\_S | ON\* | NULL | Signal input for the attribute sp\_gradient\_up |
| SP\_GRADIENT\_DOWN\_S | ON\* | NULL | Signal input for the attribute sp\_'gradient\_down |
| SP\_TRACKING\_ON\_S | ON\* | NULL | Signal input for the attribute sp\_tracking\_on |
| SP\_FAST\_MODE\_ON\_S | ON\* | NULL | Signal input for the attribute sp\_fast\_mode\_on |
| SP\_INPUT\_SIGN | ON | 'OV' | Name of the input signal (analog) |
| SP\_OUTPUT\_SIGN | ON | \* | Name of the output signal (analog) |
| SP\_MINMAX\_ON Min&Max limits used? | DO | F | Logical input, which turns on the output limiting between min and max values |
| SP\_MIN Minimum value | DO | 0.0 | Output minimum value (is sp\_minmax\_on is T) |
| SP\_MAX Maximum value | DO | 100.0 | Output maximum value (is sp\_minmax\_on is T) |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Set_point.html#Contents)  
[**User interface components**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/auuse.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**SET\_BINARY**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\F85F9CF5.tmp | Setpoint Binary (Binary Setpoint) |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Set_binary.html#Introduction)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Set_binary.html#Attributes)  
[**Binary automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**Introduction**

The SET\_BINARY module is used to give binary setpoint values. Normally the value is given by the attribute SB\_VALUE. In tracking mode, the output follows the SB\_INPUT\_SIGN value instead.  
[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Set_binary.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| SB\_VALUE Logical value | LO | F | Manually given logical value |
| SB\_INPUT\_SIGN | ON\* | NULL | Name of the input signal (binary) |
| SB\_TRACKING\_ON Tracking on? | LO | F | Logical input which turns the set\_binary to tracking\_mode |
| SB\_TRACKING\_ON\_S | ON\* | NULL | Signal input for the attribute "tracking\_on" |
| SB\_OUTPUT\_SIGN | ON | \* | Name of the output signal (binary) |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/Set_binary.html#Contents)  
[**Binary automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/aubin.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**Error messages**

The complete list of possible error messages and their explanations are listed. If the error message is specially connected to some module, then it is also included in the description of that module.

The error messages can appear in three stages of the design procedure:

* when the user creates the module
* when the user starts the simulation and the system initializes the defined experiment
* during the simulation

**Errors during the definition of the module**

In the definition stage of the module the following checks are made:

* All modules, to which the user refers to when defining the new component are defined earlier in the database. It is also checked that the module type of the given module name is correct. For example, the user cannot connect the binary signal to the setpoint signal of the controller.
* All obligatory attributes of the module are defined correctly.
* Checking that the user given values are sensible (for some module types). E.g. the upper limit of the measurement range cannot be smaller than the lower limit.

**Errors during the initialization of the experiment**

In the initialization stage of the simulation experiment, the correctness of the connections between the components is checked, for example that signal is not an output of two modules.

**Errors during the simulation**

During the simulation of the experiment, the checking for correctness of different parameters is carried out. Some examples of these kinds of checks are mentioned below:

* The denominator of the divider module should not be zero
* The upper limit of the limiter module should be larger than lower limit
* The value of hysteresis of the hysteresis module should always be positive

These kinds of checks are also done in the initialization stage of the experiment, but because the checked parameters can vary with some signals, it is necessary to check these parameters also during the simulation. If the user gets these kinds of error messages during the simulation, the user should check simulation definitions carefully. The results the user gets after this kind of error message are not necessarily reliable.

Table: Error codes in Apros Automation System

|  |  |  |
| --- | --- | --- |
| **Error code** | **Error message** | **Description of the error** |
| 4052 | CAN NOT ADD CALCULATION ORDER MODULES | Analog or binary signal CALCULATION\_ORDER modules could not be added with the command "UPDATE/ALL CALCULATION\_ ORDER C\_ANALOG\_ORDER EXIT". Simulation should not be started without these modules. |
| 4053 | DO NOT START SIMULATION | Signal address calculation failed or CALCULATION\_ORDER modules could not be added. The automation system will not function properly and the simulation should not be started. In address calculation failed, the number of the fatal errors is also printed. |
| 4081 | NO OUTPUT SIGNAL DEFINED FOR | Component in preparation without an output signal. Since output signals are automatically created when a module is added, there might be an error in the generic component level. |
| 4082 | NO INPUT SIGNAL DEFINED FOR | Component in preparation without an input signal. |
| 4084 | NO STATE SIGNAL DEFINED FOR | Command module of sequential program or on/off device controller in preparation without a state signal. Since output signals are automatically created when a module is added, there might be an error in the generic component level. |
| 4088 | NO STEPS DEFINED FOR | Command module of the sequential program in preparation without any steps. |
| 4089 | NO ERROR SIGNAL DEFINED FOR | Command module of the sequential program in preparation without a control time exceeded signal. |
| 4092 | NUMBER OF DEVICE CONTROLLERS WRONG | The number of signals connected to the device controllers does not match the number of device controllers in CHANGE\_ OVER module. |
| 4093 | NUMBER OF PRESELECTION SIGNALS WRONG | The number of preselection signals connected to the CHANGE\_OVER module does not match the number of device controllers. |
| 4096 | COMMON ADDRESS NOT FOUND FOR | Calculation of variable common index failed. |
| 4102 | CHECK THE DEFINITIONS OF MAX SELECTOR MODULE | The dimension of input signal array is not equal to the dimension of the limiting channel indication signal array. |
| 4103 | MODULE HAS NO DEFINED INPUTS | No input signals for a MIN\_ or MAX\_SELECTOR. |
| 4104 | CHECK THE DEFINITIONS OF MIN SELECTOR MODULE | The dimension of the input signal array is not equal to the dimension of the limiting channel indication signal array. |
| 4105 | MAX SELECTOR HAS NOT ENOUGH INDICATION SIGNALS | More input signals than indication signals. |
| 4106 | MAX SELECTOR HAS TOO MANY INDICATION SIGNALS | More indication signals than input signals. |
| 4107 | MIN SELECTOR HAS NOT ENOUGH INDICATION SIGNALS | More input signals than indication signals. |
| 4108 | MIN SELECTOR HAS TOO MANY INDICATION SIGNALS | More indication signals than input signals. |
| 4121 | NAME REFERENCE PAIR INCORRECT | The common index can not be calculated from the name reference pair. |
| 4122 | NAME REFERENCE PAIR ATTRIBUTE TYPE WRONG | Name index pair does not have the correct data type. |
| 4131 | CHECK THE PARAMETERS OF THE CONTROLLER | Integration or derivation time or derivation gain wrong. |
| 4132 | CHECK THE OUTPUT LIMITS OF THE CONTROLLER | The upper limit of controller output less than the lower limit. |
| 4133 | CHECK THE MEASUREMENT RANGE OF | The upper limit of measurement range less than the lower limit. |
| 4134 | CHECK THE FILTERING TIME CONSTANT OF | Measurement filtering time constant less or equal to 0. |
| 4135 | CHECK THE HYSTERESIS OF | AB- converter hysteresis < 0 |
| 4136 | CHECK THE LIMITS OF THE LIMITER | Upper limit less than the lower limit. |
| 4137 | CHECK THE DEAD BAND VALUE OF | Negative dead band value |
| 4138 | MEASUREMENT SIGNAL NOT DEFINED FOR | Controller measurement signal is not defined. |
| 4155 | SETPOINTG SIGNAL NOT DEFINED FOR | Controller set point signal not defined. |
| 4161 | OUTPUT ADDRESS NOT CORRECT | Output address of PLACE\_VALUE or \_LOGICAL is not correct. Probably an error in the subroutine AUPLACE or AUPLAL which create these modules. |
| 4162 | NO INPUT SIGNAL FOR | input signal is not defined for PLACE\_VALUE or \_LOGICALmodule. Probably an error in the subroutin AUPLAC or AULAL which create these modules. |
| 4181 | NUMBER OF SIGBNALS EXCEEDED | There exists more than 100 signals in one control circuit. Problems in automatic naming of signals. |
| 4202 | SECOND INPUT VALUE INCORRECT | Second input name reference pair of DIFFERENCE\_MEASUREMENT does not point to a real or double precision value. |
| 4203 | FIRST INPUT VALUE INCORRECT | Name reference pair of a calculation level measurement module does not point to a real or double precision value. |
| 4210 | SIGNAL DOES NOT EXIST | PLACE\_VALUE input string is not a valid signal name. |
| 4211 | Signal connection failed. No place\_value added | PLACE\_VALUE can not be added. It is needed to transmit value of (secondary) input analog signal. The reason for failure is that the tried name is already reserved. The name of the component (whose input the signal is) is too long to create the place\_logical/value unambiguously. Give a new, shorter [**name**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/guides/autgui/autgui.htm#Naming)! |
| 4212 | INPUT SIGNAL IS NOT AN ANALOG\_SIGNAL | PLACE\_VALUE input signal is not an ANALOG\_SIGNAL. |
| 4230 | SIGNAL DOES NOT EXIST | Generic component input string is not a valid signal name. |
| 4240 | SIGNAL DOES NOT EXIST | PLACE\_LOGICAL input string is not a valid signal name. |
| 4241 | Signal connection failed. No place\_logical added | PLACE\_LOGICAL can not be added. It is needed to transmit value of (secondary) input binary signal. The reason for failure is that the tried name is already reserved. The name of the component (whose input the signal is) is too long to create the place\_logical/value unambiguously. Give a new, shorter [**name**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/guides/autgui/autgui.htm#Naming)! |
| 4242 | INPUT SIGNAL IS NOT A BINARY\_SIGNAL | PLACE\_LOGICAL input signal is not a BINARY\_SIGNAL. |
| 4250 | TOO MANY STEPS CONNECTED TO | More than 1 step connected to a sequence control step. |
| 4260 | LIMITS INCORRECT | DEVICE\_CONTROLLER\_CONT: The lower limit is bigger than the higher limit  ACTUATOR: The limits related to input range, output, HH, H, L, LL are given incorrectly. For example, if the user gives an output minimum that is bigger than the maximum, the calculation replaces the user given value to be equal with the maximum value. |
| 4270 | nonstandard name | COMPONENT NAME DOES NOT MATCH SYNTAX STRING IN ARRAY 'aapsyn`. |
| 4261 | Library name not given for EXT\_AUT | No external library name is provided by the user. |
| 4262 | Routine name not given for EXT\_AUT | No external routine name is provided by the user. |
| 4263 4265 | Library not found for EXT\_AUT | The external library suggested by the user was not found by Apros. |
| 4264 4266 | Routine not found for EXT\_AUT | The external library suggested by the user does not containt the routine name. |
| 4330 | modify calc\_order\_ana\_dim VD\_SCALE\_N and | Messages 4330 and 4331 are given consecutively. The user must enlarge the dimensions of the analog calculation order tables. Give the following command in the Apros command window: \* show show calc\_order\_ana\_dim From this you should check the value of SCALE\_N. Next give the command: \* modi calc\_order\_ana\_dim vd\_scale\_n X, where X is some number larger than the SCALE\_N show previously \* arrange\_commons \* prepare (give this command after the steps in error message 4331)  If the error message comes again, give the modi and arrange\_commons commands again using a larger X |
| 4331 | calc\_order\_ana\_dim2 VD\_SCALE\_J larger | See above.  Give the following command in the Apros command window: \* show show calc\_order\_ana\_dim2 From this you should check the value of SCALE\_J. Next give the command: \* modi calc\_order\_ana\_dim vd\_scale\_j X, where X is some number larger than the SCALE\_J show previously \* arrange\_commons \* prepare (steps in error message 4330 should be also done before preparation) |
| 4332 | modify calc\_order\_bin\_dim vd\_scale\_n and | Messages 4332 and 4333 are given consecutively. The user must enlarge the dimensions of the binary calculation order tables. |
| 4333 | calc\_order\_bin\_dim2 vd\_scale\_j larger | See above. |
| 4340 | ARRAY IS NOT DEFINED IN DATABASE | An attempt was made to calculate the dimensions of a variable that is not defined in the database. |
| 4350 | NAME IS NOT MODULETYPE | MEASUREMENT\_CONNECTION input data incorrect. If the simulation database does not contain all module types that are used in the file AUTO\_PAR.QUE this error is not fatal. |
| 4351 | ATTRIBUTE DOES NOT BELONG TO MODULETYPE MEASUREMENT\_CONNECTION | input data incorrect. If the simulation database does not contain allo module types that are used in file AUTO\_PAR.QUE this error is not fatal. |
| 4360 | CAN'T FIND MODULETYPE FOR | Either measurement module type or module type of measured object is not defined in the MEASUREMENT\_CONNECTION module. |
| 4361 | CAN'T FIND MEASUREMENT\_ATTRIBUTE MODULE FOR | No MEASUREMENT\_CONNECTION data for this module type. |
| 4362 | CAN'T FIND ATTRIBUTE FOR | Attribute not defined in the MEASUREMENT\_CONNECTION data. |
| 4370 | Wrong actuator type with the controlled device | Error: The actuator of the selected type cannot be used with the device it is connected to. |
| 4371 | Actuator is not properly connected | Warning: The actuator does not have a proper connection to any process device. |
| 4373 | Driving time of the valve is zero | Error: User has not given a positive driving time for a valve. |
| 4375 | Renewing the command of the actuator: | Error: Actuator of the type 1 gives this error message when the open/close command is not reset inside the user given command renewing interval (act\_cmd\_renew\_interval) and the command is renewed. |
| 4376 | Simultaneous open and close commands in actuator: | Error: Open/Close or Start/Stop commands are given simultaneously. In this case, the close/stop command is accepted. |
| 4444 | ENDLESS LOOP IN AUTOMATION SYSTEM |  |
| 4445 | PROBLEMS IN CALCULATION ORDER | There exists a component for which the output signal must be calculated although all of its input signals have not been calculated earlier during the current iteration loop. |
| 4446 | Arithmetic loop | Most of the automation components are sorted, i.e. arranged into a calculation order automatically. See more about calculation principles [**here**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/guides/autgui/autgui.htm#Calculation_principles). An arithmetic loop may be found, when the components are sorted into the calculation order in the preparation phase. It means, that an output of a component appears (maybe after many components between) as an input of the same module. In this situation, it is hard to reason the best order, so the calculation chain is cut randomly. If problems arise because of the loop, the user can place a set\_point or a set\_binary in tracking mode into an appropriate place of the loop. Set\_point and set\_binary components cut the loop, because their input is transmitted as an output with a delay of one calculation step. |
| 4650, 4651 | Output value of flip-flop different from value of output signal: modify correct value to flip-flop | The output value of the flip-flop component (attribute FF\_OUTPUT\_VALUE) is different from the value of the output signal of the flip-flop. The user should check the correct output value (TRUE or FALSE) and modify it to the flip-flop. The value of the output signal is automatically updated when the output value of the flip-flop is modified. |
| 4652, 4653 | Output value of flip-flop = value of inverse output signal: modify correct value to flip-flop | The output value of the flip-flop component (attribute FF\_OUTPUT\_VALUE) is the same as the value of the inverse output signal of the flip-flop. The value of the inverse output signal signal should always be the inverse of the output value of the flip-flop. The user should check the correct output value (TRUE or FALSE) and modify it to the flip-flop. The value of the inverse output signal is automatically updated when the output value of the flip-flop is modified. |

[**Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/Autref/auref.html#Automation)

[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**ADDER**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\8EE9DE9E.tmp | Adder-3 (Adder) |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\490E245C.tmp | Adder-5 (Adder) |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\87A076CA.tmp | Adder-5x5 (Adder2) |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/Adder.html#Introduction)  
[**Structure**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/Adder.html#Structure)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/Adder.html#Attributes)  
[**Analog automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/auana.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**Introduction**

The ADDER module can be used for the adding or subtracting of analog signals. The output signal is calculated from the equation:

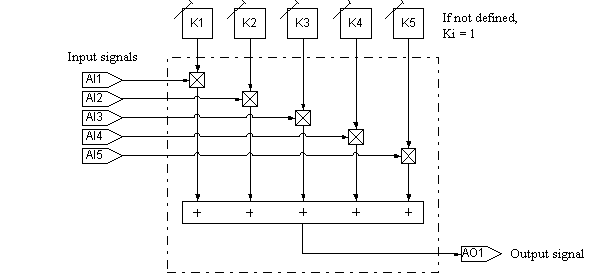
AO1 = K1\*AI1 + K2\*AI2 + ... + K29\*AI29 + K30\*AI30

The values of all defined input signals are multiplied by a given coefficient before they are added. The coefficients of the input signals can depend on analog signals.

The maximum number of input signals, that one ADDER can handle is 30. The bias values, that have no corresponding input signal, have no effect to the calculation.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/Adder.html#Contents)

**Structure**

*Figure 1. Logic behind the ADDER*

Consider an ADDER module connected in an automation model. The created structure is shown in Figure 1 and the relevant parameters for this kind of a logical structure are shown in Table 1.

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Relevant parameters for an ADDER* | | | |
| **Attribute Property name** | **Type** | **Value** | **Note** |
| ADDER\_COEFFICIENT Multiplication coefficient for the input | DO(30) | 1.0 | Multiplication coefficient for each defined input signal. |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/Adder.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 2. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| ADDER\_COEFFICIENT Multiplication coefficient for the input | DO(30) | 1.0 | Multiplication coefficient for each defined input signal |
| ADDER\_COEFFICIENT\_S | ON\*(30) | NULL | Signal input for attribute adder\_coefficient |
| ADDER\_INPUT\_SIGN | ON(30) | 'OV' | Input signals (analog), a maximum 30 of input signals can be defined for one module, non defined input signals have no effect to the calculation |
| ADDER\_OUTPUT\_SIGN | ON | \* | Output signal (analog) |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/Adder.html#Contents)  
[**Analog automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/auana.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

**PI\_CONTROLLER**

|  |  |
| --- | --- |
| **Symbol** | **User interface name** |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\A3285BB6.tmp | PI-controller |
| C:\Users\moshiur\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\A0835934.tmp | PI-controller S (PI-controller) |

**Contents**

[**Introduction**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/PI_controller.html#Introduction)  
[**Structure**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/PI_controller.html#Structure)  
[**Attributes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/PI_controller.html#Attributes)  
[**Operation modes**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/PI_controller.html#Operation_modes)  
[**Error messages**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/PI_controller.html#Error_messages)  
[**Analog automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/auana.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)

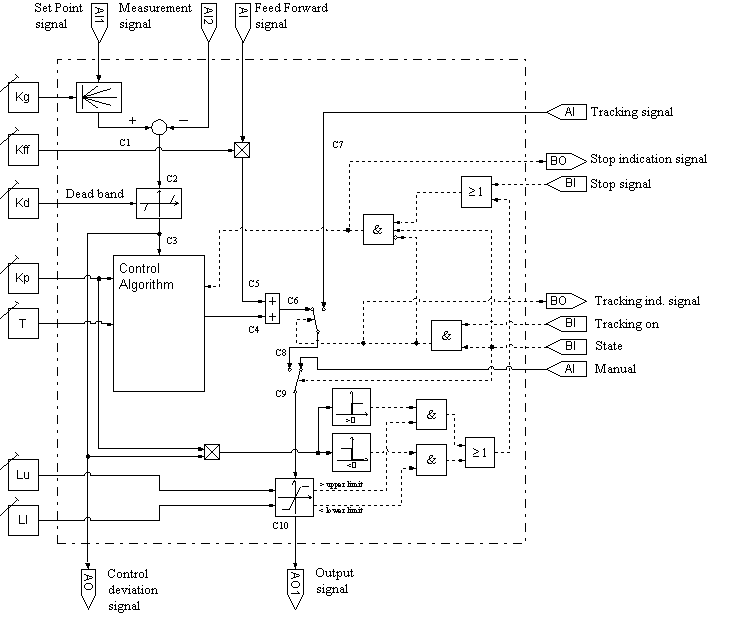
**Introduction**

**Note!** It is recommended to use module type [**CONTROLLER**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/Controller.html) (in versions 5.04->) instead of other controller module types, if possible. To define controller as PI-controller, you must set attributes CON\_P\_USED and CON\_I\_USED as T, and CON\_D\_USED as F.

The PI\_CONTROLLER module can be used in a closed loop control circuit to perform a simple PI­-control algorithm.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/PI_controller.html#Contents)

**Structure**

*Figure 1. Logic behind the PI\_CONTROLLER*

Consider a PI\_CONTROLLER module connected in an automation model. The created structure is shown in Figure 1 and the relevant parameters for this kind of a logical structure are listed in Table 1

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 1. Relevant parameters for a PI\_CONTROLLER* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Value** | **Note** |
| PI\_STATE Controller in AUTO mode? | LO | T |  |
| PI\_GAIN Gain | DO | 0.01 |  |
| PI\_INTEGRATION\_TIME Integration time | DO | 1000 |  |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/PI_controller.html#Contents)

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 2. Attributes, types, default values and descriptions* | | | |
| **ATTRIBUTE** **Property name** | **Type** | **Default value** | **Description** |
| PI\_STATE Controller in AUTO mode? | LO | T | Logical value, when TRUE controller is in auto mode |
| PI\_MANUAL Output value | DO |  | Value of manual demand, in auto mode this value follows the output of the controller |
| PI\_GAIN Gain | DO | 0.0 | Value of the gain of the controller (Kp) |
| PI\_INTEGRATION\_TIME Integration time | DO | 1000.0 | sec, Value of the integration time of the controller (Ti, seconds) |
| PI\_FF\_COEFFICIENT Gain for feedforward control | DO | 0.0 | Value of the gain of the feed forward signal (Kff) |
| PI\_SETPOINT\_GRAD Max. gradient for set point change | DO | 1000000.0 | 1/min, Value of the set point gradient, (Kg) unit is 1/min |
| PI\_DEAD\_BAND\_VALUE Dead band value | DO | 0.0 | Value of the dead band of the control deviation (Kd) |
| PI\_OUTPUT\_UPPER\_LIMIT High limit | DO | 1.0 | Upper limit of the controller output (Lu) |
| PI\_OUTPUT\_LOWER\_LIMIT Low limit | DO | 0.0 | Lower limit of the controller output (Ld) |
| PI\_VALUES Values of controller internal matrix | DO(3,6) |  | Real matrix showing the values of some inside variables of the controller; (1) set point after gradient, ; (2) control deviation, ; (3) output of the control algorithm, ; (4) output of the controller |
| PI\_STATE\_S | ON\* | NULL | Signal input of the attribute 'state' |
| PI\_GAIN\_S | ON\* | NULL | Signal input of the attribute 'gain' |
| PI\_INTEGRATION\_TIME\_S | ON\* | NULL | Signal input of the attribute 'PI\_INTEGRATION\_TIME' |
| PI\_FF\_COEFFICIENT\_S | ON\* | NULL | Signal input of the attribute 'PI\_FF\_COEFFICIENT' |
| PI\_SETPOINT\_SIGN | ON | 'OV' | Name of the set point signal (analog) |
| PI\_MEASUREMENT\_SIGN | ON | 'OV' | Name of the measurement signal (analog) |
| PI\_FEED\_FORWARD\_SIGN | ON | NULL | Name of the feed forward signal (analog) |
| PI\_TRACKING\_SIGN | ON | NULL | Name of the tracking signal (analog) |
| PI\_TRACKING\_ON\_SIGN | ON | NULL | Name of the tracking mode signal (binary); when this signal is TRUE, the output of the controller follows the tracking signal and output of the control algorithm follows the output of the controller |
| PI\_STOP\_SIGN | ON | NULL | Name of the stop signal (binary input), when is signal is TRUE, the integration part of the controller will be stopped |
| PI\_TRACKING\_IND\_SIGN | ON | \* | Name of the output signal (binary). ; This signal is TRUE, when controller is in tracking mode |
| PI\_STOP\_IND\_SIGN | ON | \* | Name of the output binary signal. ; This signal is TRUE, when controller is in stop mode |
| PI\_CONT\_DEVIATION\_SIGN | ON | \* | Name of the output signal (analog), value of this signal equals to the control deviation of the controller |
| PI\_OUTPUT\_SIGN | ON | \* | Name of the output signal of controller (analog) |
| PI\_STATE\_IND\_SIGN | ON | \* | Binary output indicating the state of the controller: F = manual mode ; T = automatic mode |
| PI\_NEW New controller module | LO | T/F | Logical attribute, should not be modified by the user |

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/PI_controller.html#Contents)

**Operation modes**

The controller have three different operation modes:

* manual mode,
* normal control mode (automatic mode),
* tracking mode,

The actual operation mode is determined by the 'state' input and the 'tracking\_on\_sign'input signal.

In manual mode (state input 'False') the output of the controller is determined by the operator. Output can be changed manually by modifying the value of the attribute 'manual'.

In normal control mode (state input 'True' and tracking\_on\_sign in 'False' state), the output of the controller is calculated based on the control deviation of the controller. The transfer function is in the case of PI\_controller:

O(s) = Kp\*((1 + 1 / (Ti\*s )) \* E(s) + Kff\*F(s)

O(s)   Output of the controller  
E(s)   Control deviation of the controller  
F(s)   Feed forward signal of the controller  
Kp    Controller gain  
Ti      Integration time (sec)  
Kff    Feed forward coefficient

If the feed forward signal has not been defined, the value 0.0 is used. The control deviation of the controller is calculated as:

error = set point ­ measurement

It is possible to define the gradient limiter for the actual set point (setpoint\_grad) and also a dead band for the control deviation (dead\_band\_value).

Normal operation of the controller is such that the output of the controller will increase, if the control deviation is positive (set point is bigger than measurement). If inverse operation is needed, the gain of the controller should be defined negative (< 0).

The effect of the integration part of the the controller can be stopped by binary signal stop\_sign. When the stop\_sign is in 'True' state, the controller will behave like P­-controller. Integration part will be stopped automatically if the output of the controller reaches either of the control limits. The stop mode is indicated by binary signal 'stop\_ind\_sign'.

In tracking mode (state input 'True' and tracking\_on\_sign in 'True' state) the output of the controller is following the value of tracking signal (tracking\_sign, analog signal)and control deviation or feed forward signal have no effect to it.

In each of the three operation modes, the output of the controller is limited between the lower and upper limits.

It is also possible to define the tuning parameters of the controller (Kp, Ti) and the feed forward coefficient (Kff) to be dependent on some analog signal.

The change from manual to automatic mode is jumpless. During the manual mode the output of the control algorithm is updated from the given manual command and the set point after the gradient limiter is following the measurement signal.

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/PI_controller.html#Contents)

**Error messages**

**4131 Check the parameters of the controller:** The integration time of the controller is negative

**4132 Check the output limits of the controller:** The upper limit of the controller outputs smaller than the lower limit

**4155 Setpoint signal not defined for...:** Set point signal of the controller is missing

[**To contents of this component**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/PI_controller.html#Contents)  
[**Analog automation**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/refs/autref/auana.html)  
[**Main Table of Contents**](http://127.0.0.1:57778/help/topic/fi.vtt.apros.manual/doc-user/html/Combustion/index.htm)