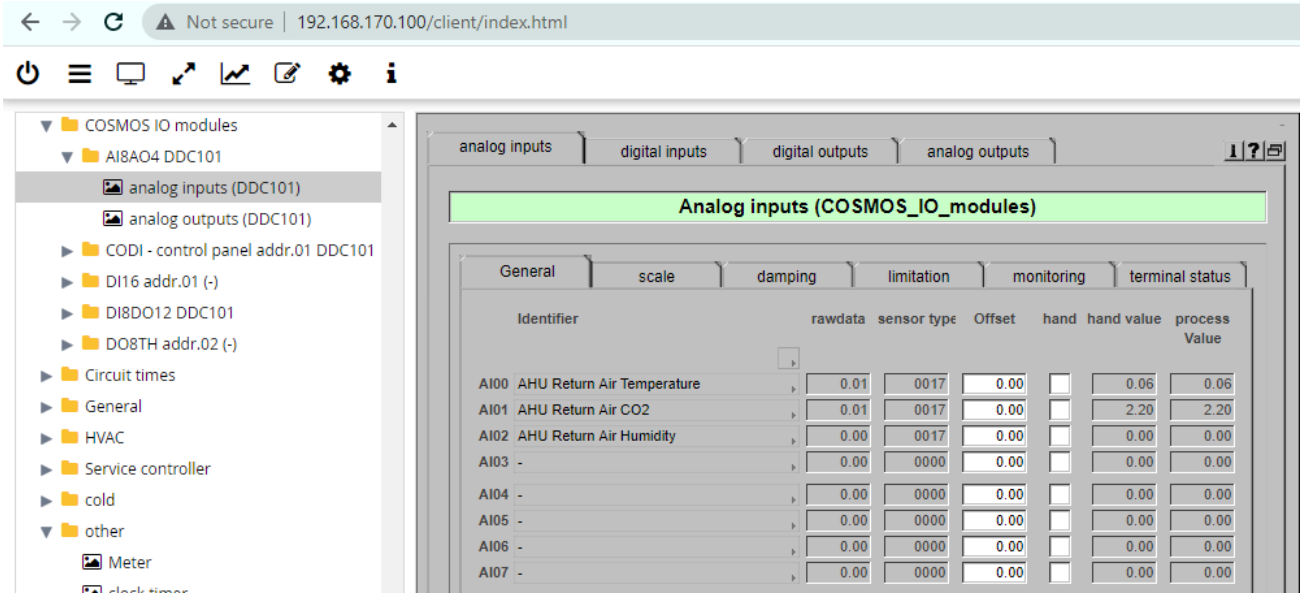
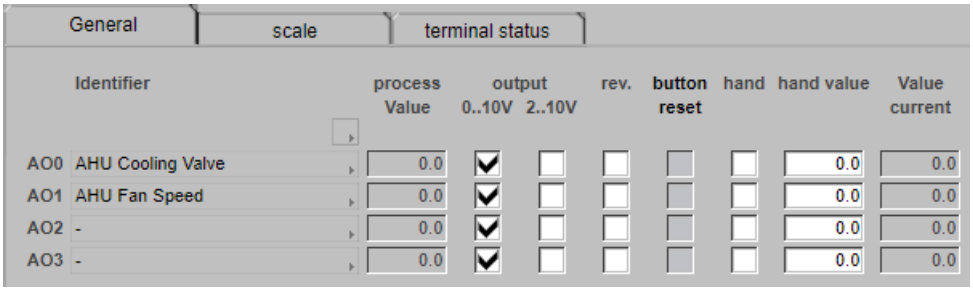
TT210902 – OPENview - Analog IO Settings

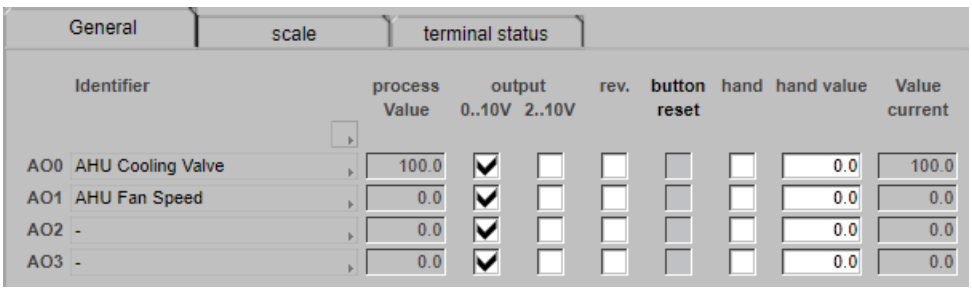
1. In this document, we will show you the settings for analog input and output one by one. Start OPENview in a HTML5 browser, e.g. Chrome. From the menu on the left, click “COSMOS IO modules” and then the OPEN 600 internal IO “AI8AO4”.



2. Let’s take a look at the easy one, AO first. Click on “analog outputs” and you should see the screen like below. The “process value” is the value from the command (“reference”) in your FUP page (or other macro) and the “value current” is the actual output at the AO terminal.



3. In the above example, the cooling valve is closed and so the “process value” is 0. Therefore the “value current” is also 0, and so there is no voltage at the AO0 terminal (0V).
4. When the AHU turn on and the cooling valve start opening based on the temperature and setpoint, you should see both the “process value” and “value current” start going up.



5. In the above example, the cooling valve “process value” is at 100% (fully open). Therefore the “value current” is also 100, and there will be 10V DC at the AO0 terminal.
6. You should now see the AO0 LED on the controller start flashing yellow. It will start flashing 10 times, stop for a few seconds, and start flashing 10 times, and so on. This means the output voltage at the AO0 terminal is 10V. So for 25% (i.e. 2.5V), it will flash 2 times, stop for a few seconds, and flash 2 times. This allows you to roughly know the AO output without any tool.

7. You can manually set the AO output voltage by using “hand” and “hand value” settings. In the example below, AO0 is manually set (hand) to 50% (hand value), i.e. 5V at the AO0 terminal.

General | scale | terminal status

	Identifier	process Value	output 0..10V	output 2..10V	rev.	button reset	hand	hand value	Value current
AO0	AHU Cooling Valve	100.0	☑	☐	☐	☐	☑	50.0	50.0
AO1	AHU Fan Speed	0.0	☑	☐	☐	☐	☐	0.0	0.0
AO2	-	0.0	☑	☐	☐	☐	☐	0.0	0.0
AO3	-	0.0	☑	☐	☐	☐	☐	0.0	0.0

8. The “rev.” setting allows you to reverse the AO output. In the example below, the valve is fully closed (0%) in “value current”, when the “process value” is 100%. So the AO0 LED is off.

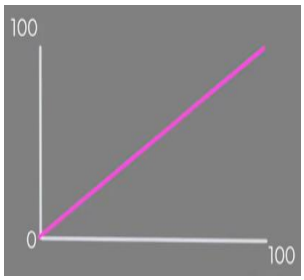
General | scale | terminal status

	Identifier	process Value	output 0..10V	output 2..10V	rev.	button reset	hand	hand value	Value current
AO0	AHU Cooling Valve	100.0	☑	☐	☑	☐	☐	0.0	0.0
AO1	AHU Fan Speed	0.0	☑	☐	☐	☐	☐	0.0	0.0
AO2	-	0.0	☑	☐	☐	☐	☐	0.0	0.0
AO3	-	0.0	☑	☐	☐	☐	☐	0.0	0.0

9. In the “Scale” tab, you can change the scale for the AO output, as well as the min/max limit value for the output. This is the default settings for your reference (when the “process value” is at 100%).

General | scale | terminal status

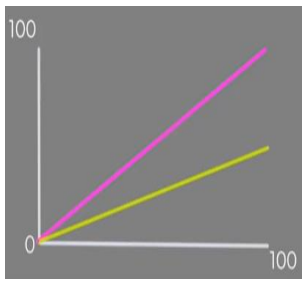
	Identifier	input Value	=	output Value	input Value	=	output Value	MIN limit	MAX limit	Value (scaled)
		---- (MIN-value) ----			---- (MAX-value) ----					
AO0	AHU Cooling Valve	0.0		0.0	100.0		100.0	0.0	100.0	100.0
AO1	AHU Fan Speed	0.0		0.0	100.0		100.0	0.0	100.0	0.0
AO2	-	0.0		0.0	100.0		100.0	0.0	100.0	0.0
AO3	-	0.0		0.0	100.0		100.0	0.0	100.0	0.0



10. When the “Output Value” (MAX-value) is set to 50. The “Value (scaled)” will become 50% (follow the yellow line in the picture), and therefore the AO0 output will be at 50%.

General | scale | terminal status

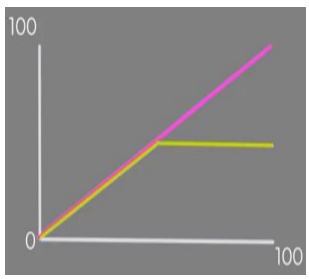
	Identifier	input Value	=	output Value	input Value	=	output Value	MIN limit	MAX limit	Value (scaled)
		---- (MIN-value) ----			---- (MAX-value) ----					
AO0	AHU Cooling Valve	0.0		0.0	100.0		50.0	0.0	100.0	50.0
AO1	AHU Fan Speed	0.0		0.0	100.0		100.0	0.0	100.0	0.0
AO2	-	0.0		0.0	100.0		100.0	0.0	100.0	0.0
AO3	-	0.0		0.0	100.0		100.0	0.0	100.0	0.0



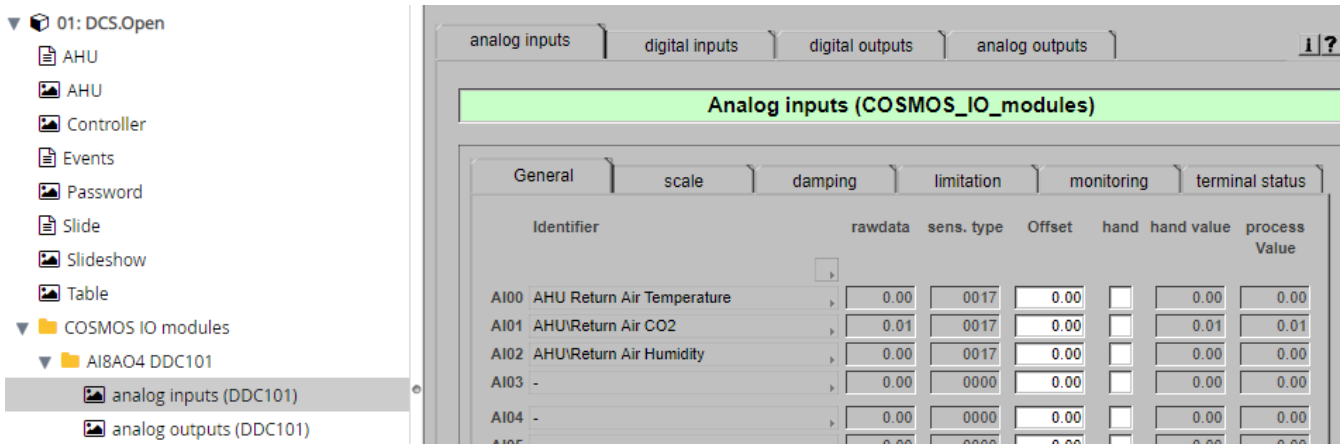
11. With the “MAX limit” setting, you can limit the maximum AO output at 50%, for example below. So with all these settings, you can easily setup the AO characteristic online using OPENview.

General | scale | terminal status

	Identifier	input Value	=	output Value	input Value	=	output Value	MIN limit	MAX limit	Value (scaled)
		---- (MIN-value) ----			---- (MAX-value) ----					
AO0	AHU Cooling Valve	0.0		0.0	100.0		100.0	0.0	50.0	50.0
AO1	AHU Fan Speed	0.0		0.0	100.0		100.0	0.0	100.0	0.0
AO2	-	0.0		0.0	100.0		100.0	0.0	100.0	0.0
AO3	-	0.0		0.0	100.0		100.0	0.0	100.0	0.0



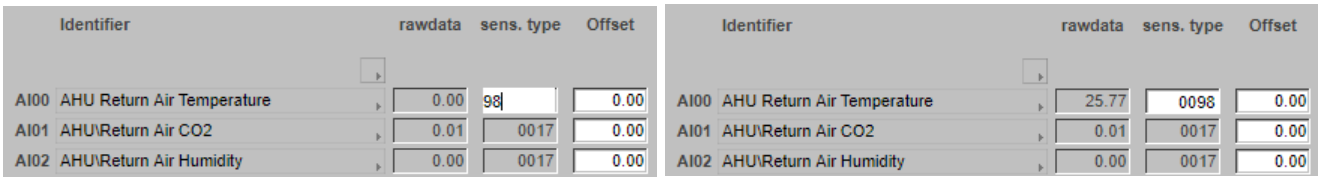
12. Now check the AI point. Please note that there is no LED on the controller for AI point.



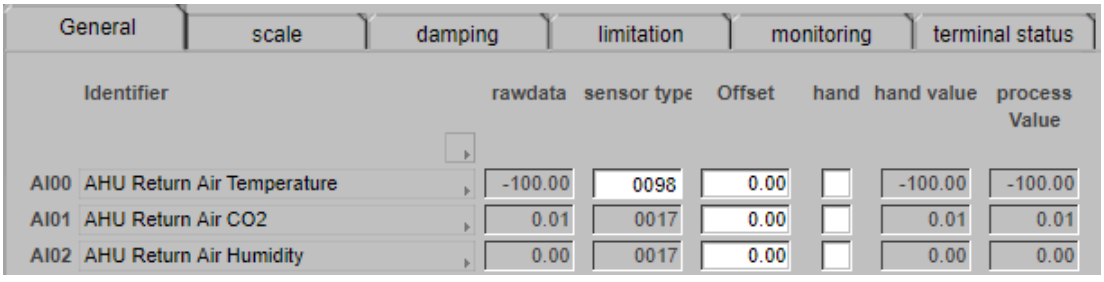
13. The first one is “rawdata”, which depends on the sensor type you set (see table below). In the above example, the type is 17 (voltage input), so it is the actual voltage at the AI terminal. If the value is 0, then maybe the “Sensor Type” is wrong, or the wiring connection is wrong.

type	ident. num	sensor	range
AS_Status_00	0	input deactivated	
AS_V_10V_2	17	voltage input	0..10V
AS_T_10mV_K_2	34	M-sensor (LM235Z)	-50°C..130°C
AS_T_Ni1000_2	50	Ni-1000 (DIN)	-50°C..650°C (2)
AS_T_Ni1000L_2	82	Ni-1000 (Tk5000)	-50°C..650°C (2)
AS_T_Pt1000_2	98	Pt-1000	-50°C..650°C (2)
AS_R_8000_2	162	resistance in ohm	0,5..10 kOhm
AS_V_percent	177	voltage input	0..100% = 0..10V

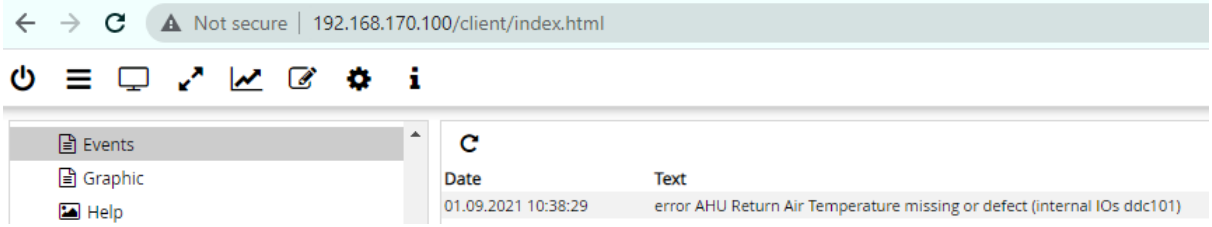
14. For AI00 (AHU Return Air Temp), the type we set in FUP is 17 (voltage input). For example, if it is connected with a PT1000 sensor, then there is no voltage and so the reading is 0. We can test it by manually changing it to 98. Click on the “Sensor Type” value, change to 98, and press “Enter”. You should now see the correct reading. Type 0 to clear the manual setting.



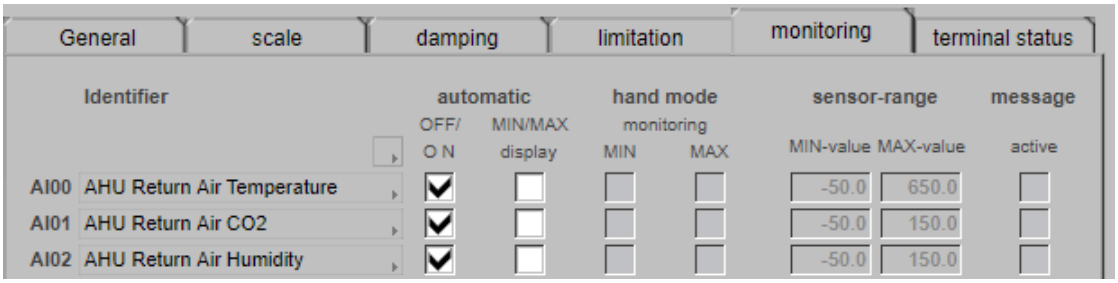
15. For sensor type 98 (PT100). if you see a big negative value in “rawdata” (like below), then most likely the sensor is not connected.



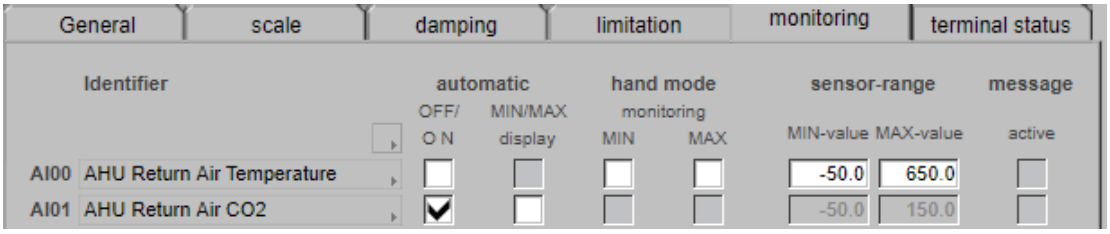
16. You will also get an alarm like below in OPENview (click the “Events” on the left).



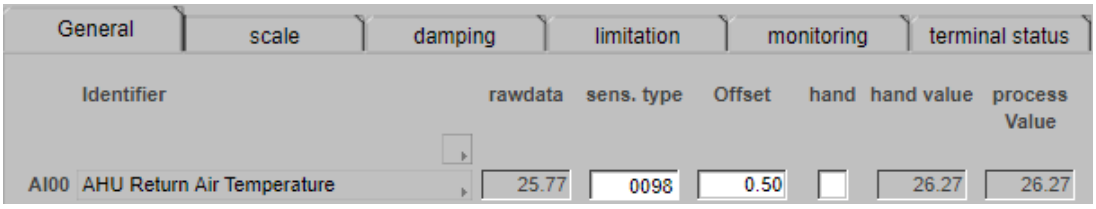
17. In the “Monitoring” tab, you can see the automatic detection of sensor failure (”automatic”) is set to “ON”. The “message” is “active” when the sensor failed or disconnected.



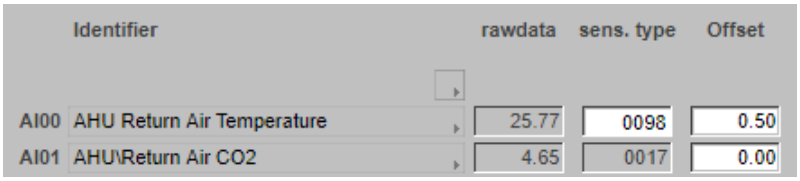
18. You can disable the alarm by setting “automatic” to “OFF” (untick it). If you want, you can set the sensor monitoring manually, by enabling “hand mode monitoring” and setting the “sensor-range” for min/max value limits.



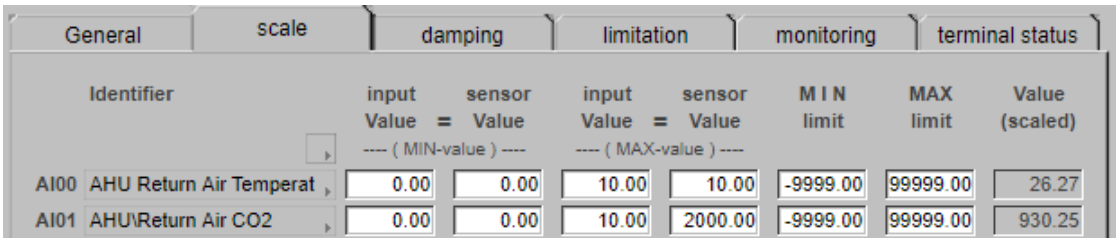
19. If the temperature reading is a bit different from the actual temperature, you can use the “Offset” to adjust it. See below, the “Process Value” is the temperature reading you will see in your FUP page, and in the graphic. Now it has an “Offset” of +0.5 from the “rawdata” reading.



20. Next example is the CO2 sensor (AI01). Most likely it is a 0-10V sensor, so the type 17 should be correct. You can see below that the “rawdata” is 4.65 meaning that the voltage measure at the AI01 terminal is 4.65V.

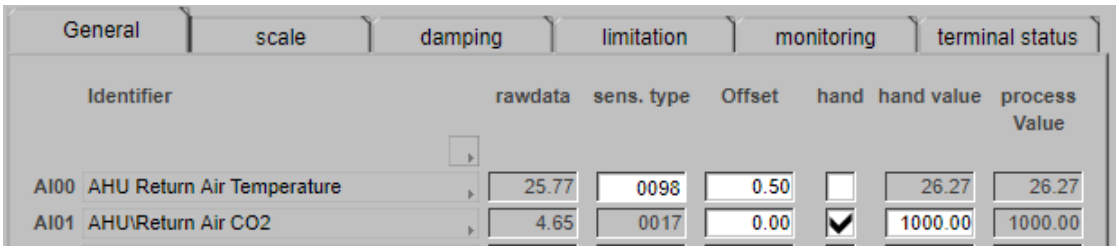


21. For 0-10V sensor, you need to set the correct sensor range according to your sensor specification. For example, if the CO2 sensor range is 0-2000 ppm, then you can set the sensor range in the “Scale” tab. Now change the “Sensor Value (Max Value)” to 2000, and you should see the correct reading in “Value (Scaled)”.

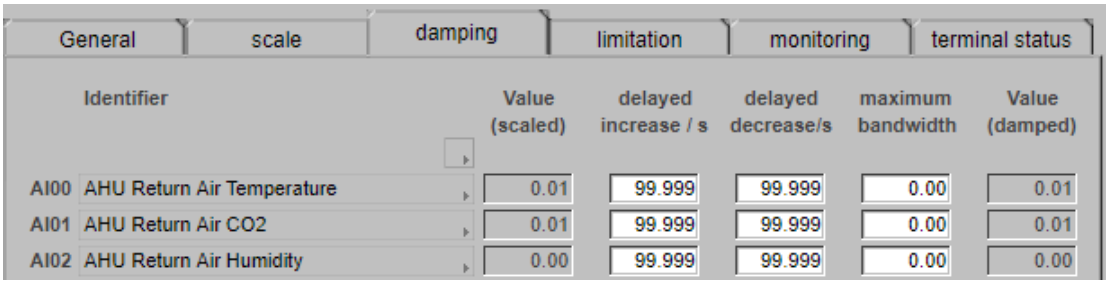


22. For some 0-10V temperature sensor, the range is -50 to 150 degree, then you need to set both the “Min Value” and “Max Value” for “Sensor Value”. You can also set the min/max limit value, so that the sensor value you got is always within the limits.

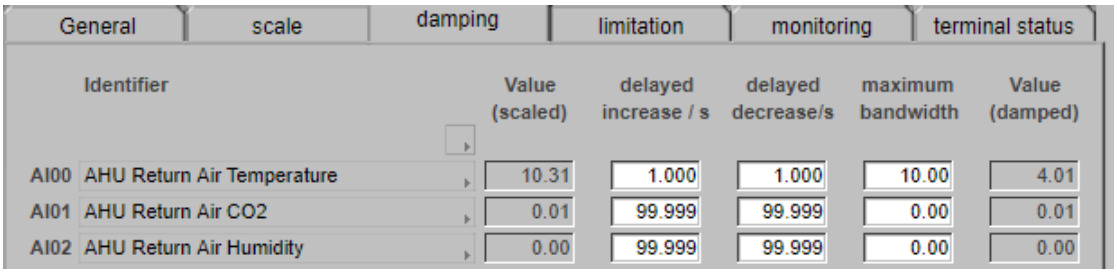
23. Sometimes you may see all the settings in the I/O module are 0, or many strange values. It is because you forget to do a “Preset” when you upload a completely new program or it’s a new controller. Please refer to TT19504 on how to “Preset” the controller
24. If you’re testing your program, and you don’t have the sensor connected, or the sensor is faulty, you can “fixed” the sensor value temporary so that you can test your program logic. Tick the “hand” checkbox, and type the value you want in the “hand value” input



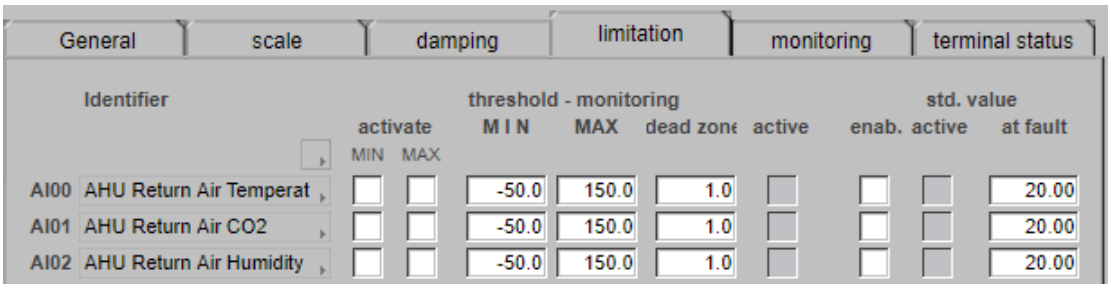
25. In the “Damping” tab, you can slow down the response of the AI value (damping), by limiting the rate of change per second and the maximum bandwidth for the changes.



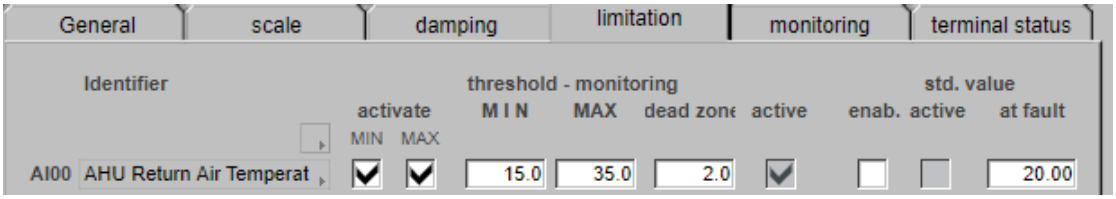
26. In the example below, when the “Value (scaled) suddenly change from 0 to 10, the “Value (damped)” will change slowing and increase by 1 every second. The value 10 in “maximum bandwidth” means that if the “Value (scaled)” change more than 10, then no damping anymore.



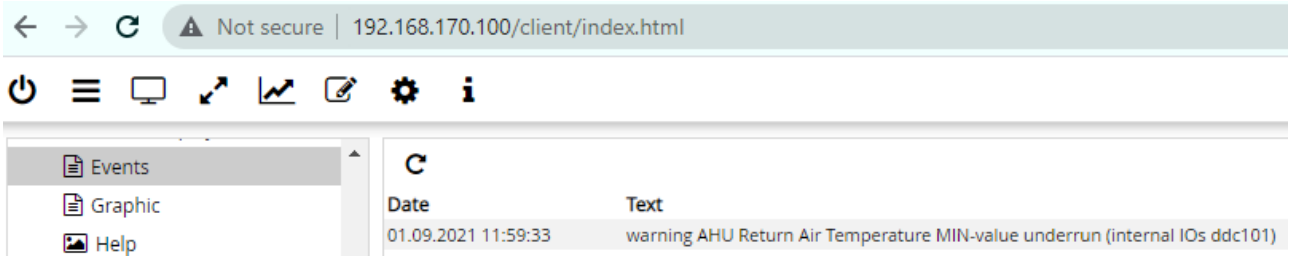
27. Finally, in the “Limitation” tab, you can set alarms when the value goes beyond the min/max limit (with dead zone), and you can set the value to “std. value” when the sensor is “at fault”.



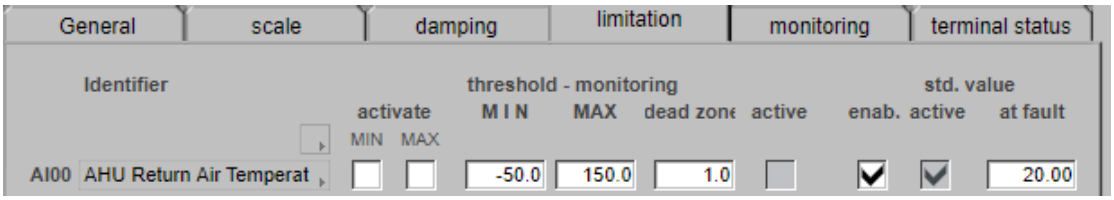
28. The example below enables the min/max alarms, with limits set to 15 and 35 degree, and a deadband of 2 degree.



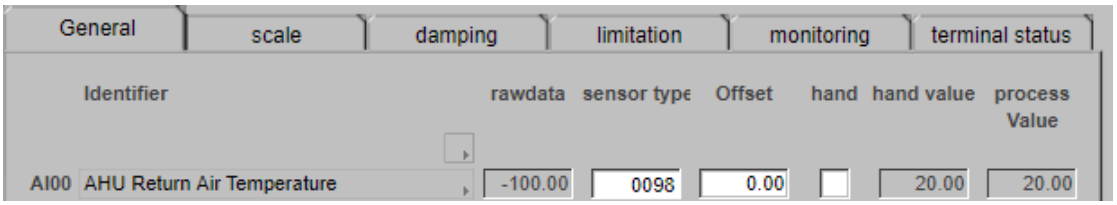
29. You will get the alarm like below when the temperature falls below 15 degree. The alarm will return to normal when the temperature goes up to 17 degree again.



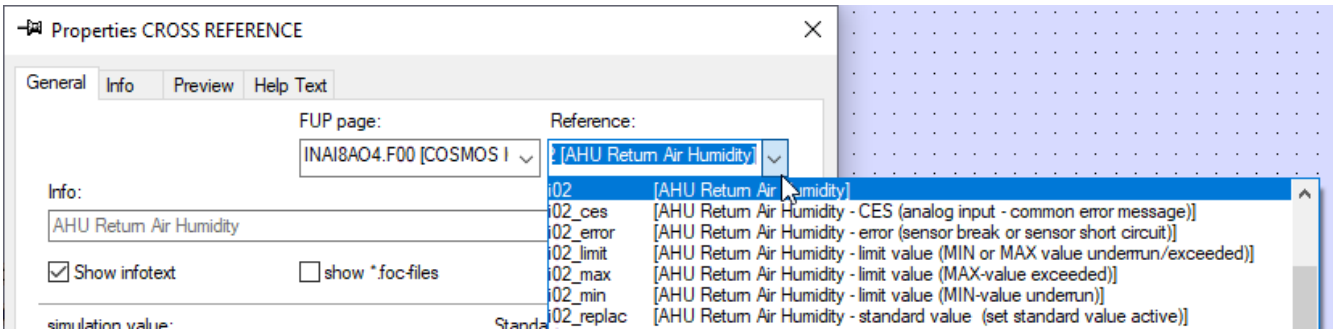
30. The “std. value”, when “enable”, will set the point value to the “at fault” value when the sensor is failed or disconnected.



31. In this example, the “process value” will be set to 20 degree, when the temperature sensor is faulty.



32. In your FUP page, you can use the follow “Reference” from the IO module, so that we can use them in your logic programming.



33. Since all the settings in the IO modules (both analog and digital) can be changed online, this provides you a very easy way to commission the system on-site, without changing your FUP program.
34. Please refer to the document TT180801 on how to create IO points in FUP.
35. If you’ve problems with getting correct values for the IO points, you can check TT190505 to troubleshoot the problems.