INTERRA

Developer of Uniqueness

iX3 4” KNX Touch Panel

Product Manual

Contents

Information in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets your specifications.
INTERRA MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR NONSTATUTORY, RELA-TED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE.

Interra disclaims all liability arising from this information and its use. Use of Interra devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Interra from any damages, claims, suits, or expenses resulting from such use. No licenses are implicitly or otherwise conveyed under any Interra intellectual rights.

Trademarks

The Interra name, logo and the Interra ITR331-XXXX iX3 device are registered trademarks of Interra Technology in Turkey and other countries.

All other trademarks mentioned herein are property of Interra Technology.

Printed on recycled paper.

TS EN ISO 9001:2008

TO OUR CUSTOMERS

One of our most important aims is to provide you with the best documentation possible to use successfully your Interra products. Focusing on this, we will keep on improving our documentation to better suit your needs. Our publications will be updated as new volumes as soon as changes are introduced.

If you have any questions or comments regarding this publication, do not hesitate to contact us:

E-Mail: info@interratechnology.com

Tel: +90 (216) 326 26 40 Fax: +90 (216) 324 25 03

Most Current Product Manual

To obtain the most up-to-date version of this product manual, please visit our Web site at:

http://www.interratechnology.com

You can determine the version of an Interra document by examining its literature number found on the bottom right corner of any page.

The first two letters of the literature are the type of document. The following numbers are the document's creation date and the last letter is the version (e.g., PM181017001A is version A of a product manual created on 17/10/18).

1. Content of The Document

This document contains Interra's ITR331–XXXX coded iX3 4" KNX Touch Panel device's electronic and all essential feature information for programming this product. In each subtitle is explained the characteristics of the device are. Modifications of the product and special change requests are only allowed in coordination with product management.

2. Product Description

Interra iX3 is a wall-mounting KNX touch panel device with an integrated temperature, humidity, brightness and air quality sensor. iX3 can have integrated AQI features. iX3 can be easily and clearly operated through the friendly interaction interface. The iX3 can control heating and cooling operating modes with 2-points, Continuous and PWM thermostat functions. Air conditioner control is available on iX3 and can control AC gateway devices. iX3 has fully touchable LCD screen with IPS-display technology which provides low energy cost and good view angle. The device provides an adjustable LCD backlight for user comfort. The product range has 4 different models with AQI and without AQI. iX3 is supported up to 13 languages such as Turkish, English, German, Russian, Arabic, French, Greek, Italian, Persian, Spanish, Brazilian, Dutch and Polish. All models can be programmable with the same ETS database, which provides efficient commissioning.

2.1. Technical Information

The following table shows the technical information of the Interra iX3 4" KNX Touch Panel.

Product Code	**ITR331–XXXX**
Power Supply	KNX Power Supply
KNX Bus Current	40mA
Push Buttons	1 x KNX Programming Button
Sensors	Temperature sensor (±0.2°C sens.) Humidity sensor (±2 %RH sens.) Air Quality Sensor (0-500 VOC Index) Brightness Sensor (Up to 1800 Lux.)
Interfaces	IPS Touch Screen
Mode of Commissioning	S-Mode
Type of Protection	IP 20
Temperature Range	Operation (– 5°C …45 °C) Storage (– 20°C …60 °C)
Maximum Air Humidity	< 90 RH
Colour	White, Black
Dimensions	84 x 100 x 8.6 mm (W x H x D)
Certification	KNX Certified
Configuration	Configuration with ETS

2.2. Models And Variations

I T R 3 3 1 – X_1 X_2 X_3 X_4

- X_4 → Colour
- X_3 → Reserved
- X_2 → Reserved
- X_1 → AQI is available or not

X_1	0	2
AQI	×	✓

Table 1: iX3 AQI Status

X_4	1	2
Colours	Black	White

Table 2: iX3 Colours

2.3. Dimensions

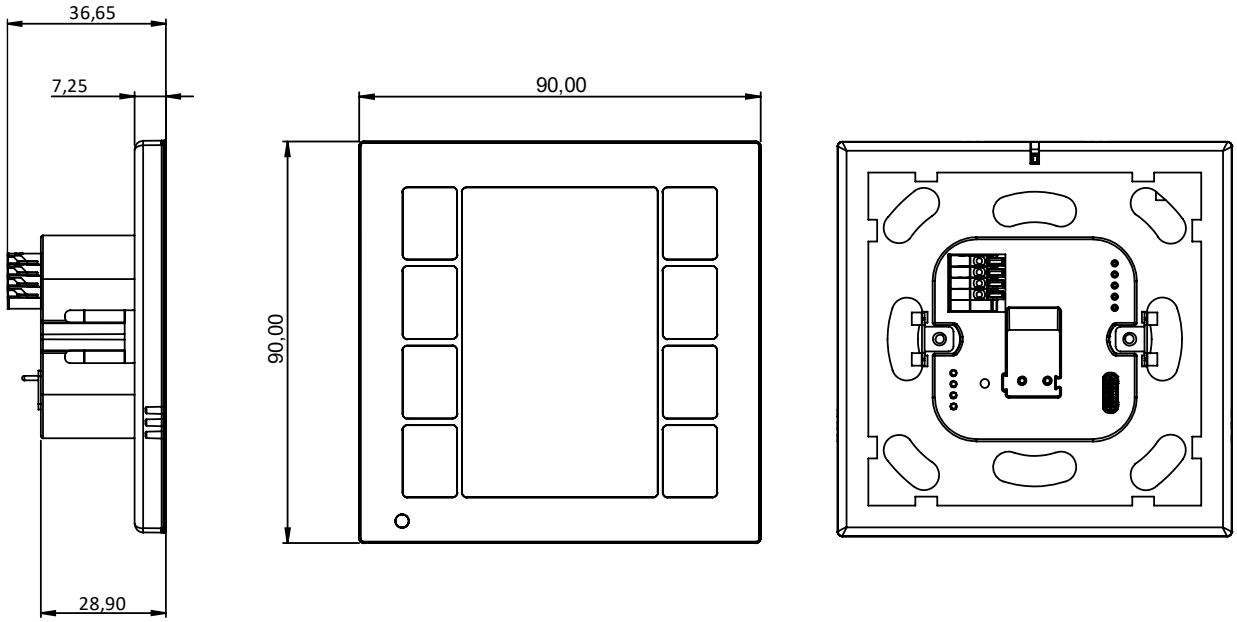


Fig. 1: Dimensions of the iX3

• All values given in the device dimensions are millimetres.

2.4. Functional Descriptions

The prominent features of the iX3 are the followings:

Up to 44 functions can be controlled separately.

- Up to 6 functions pages that can be controlled different functions.
- Switching, toggle, dimming, shutter/blinds controls, predetermined scenes by users, value functions that can send presented values, 2 channels control functions, step switching, music control and thermostat extension features are available.
- It can measure with integrated temperature, humidity, brightness and air quality (depending on the model) sensors.
- Configurable and programmable external inputs as analog or digital over ETS.
- Scenes from 1 to 64 can be specified and these scenes can be implemented by request.
- Room temperature regulation can be done with 2 –Points (Hysteresis), PWM or Continuous PI control options.
- Operating modes: comfort, standby, economy and protection.
- Automatic switching between operating modes via the weekly program.
- Enhanced and extended air conditioner control functions.
- Enhanced and extended touchable LCD screen functions. The user can be configured the function pages.
- Thermostat control, AC control, Music system control, RGB or RGBW control, Dimming control, Dimming Tuneable White control and Shutter/Blinds/Jalousie control etc. can be controlled on function pages specially designed for these functions.
- Home-page navigation function.
- Password screen is available. Up to 4 passwords can be set.
- Screen saver function is available. The users can set the screen saver. The screen saver can be digital clock. Additionally, screen saver can decrease or off the backlight for energy saving.
- The users can change the screen theme via the device. Dark and light themes are supported.
- The users can change the system language via the device.
- Display time, date, temperature and humidity.
- Logic and converter functions, AND, OR, XOR, gate forwarding, threshold comparator and conversion of different data point types.

2.5. Connection to The KNX Bus and Programming

The connection of the KNX bus line is made with the terminal block (black/red socket group) included in delivery and inserted into the slot of housing.

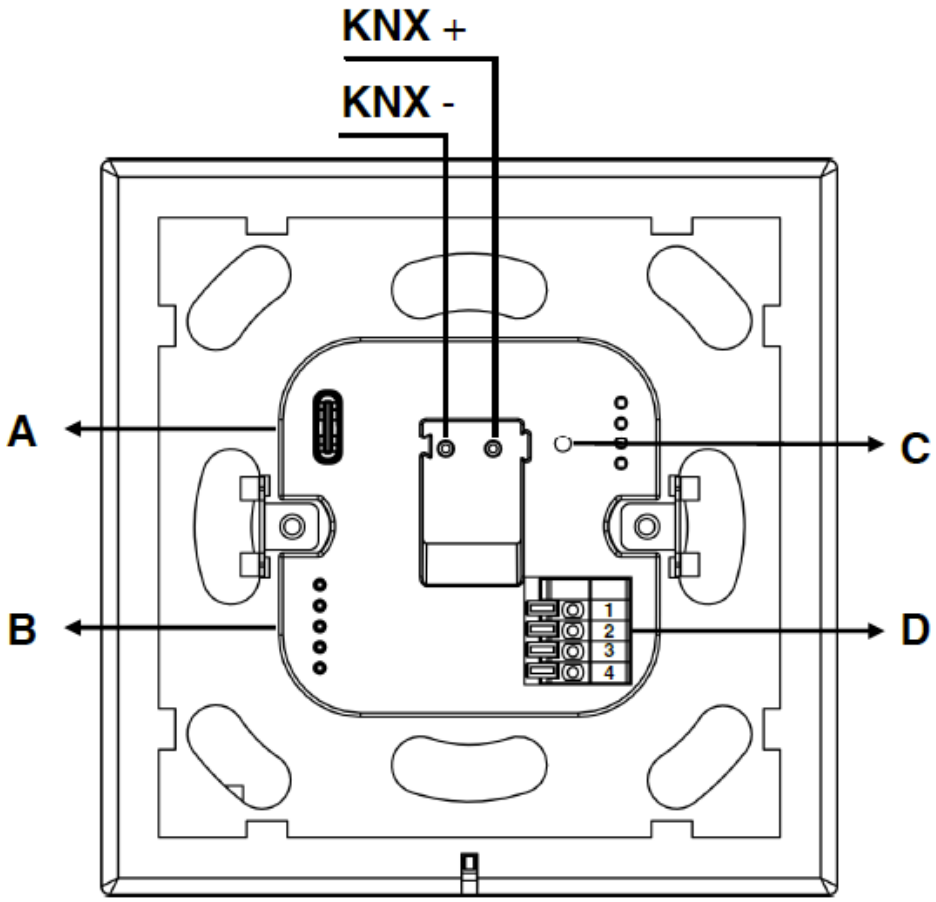


Fig. 2: Connection to KNX and Programming Button

A	USB Connection Port
B	Upgrade Port
C	KNX Programming Button
D	External Inputs

Table 3: Connection Diagram

Special Note

If the device database unloaded, you can also switch to programming mode by tapping the icon displayed on the screen.

3. Mounting

The iX3's mounting steps are described below.

Mounting

The device is suitable for use in dry interior rooms and can only be mounted on a standard-sized round or square wall flush mounting box. The iX3 should be mounted after the wall painting process is finished. Otherwise, the product's cosmetics may be damaged. The mounting steps are shown below.

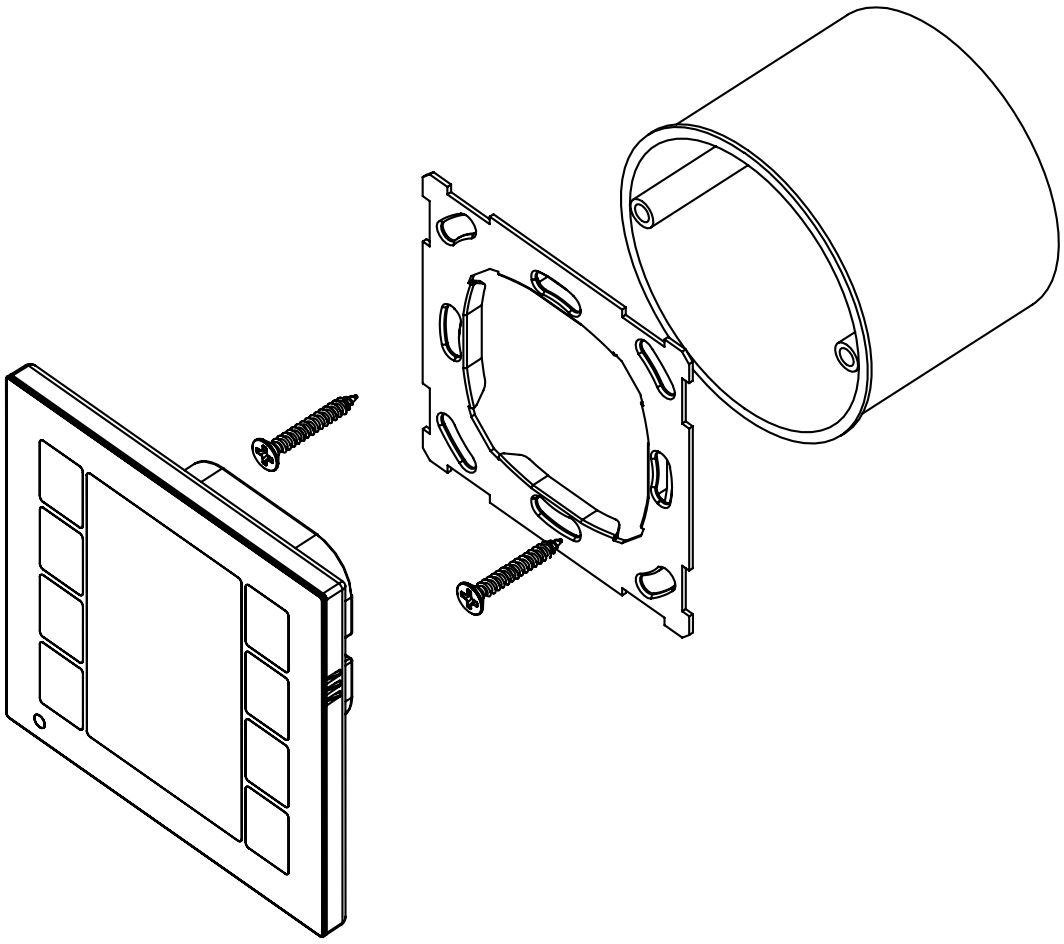


Fig. 3: Mounting the to Flush Mounting Box

1. Check the mounting box, and whether is done properly

2. Insert the metal frame of the iX3 into the wall flush mount. *

3. Connect the KNX cable and other cables (optional) to respective terminals.

4. Finally, insert the iX3 into the metal frame.

***:** Screw down strength is 1 Nm.

4. ETS Parameters

4.1. General Page

When the iX3 is attached to the project from the ETS program, a configuration setting must be made primarily before loading, depending on the model to be programmed. When entering the "GENERAL" in the parameter page, the configuration screen will appear shown above. As previously mentioned, all models can be configured via an ETS file thus the programmers can work flexibly.

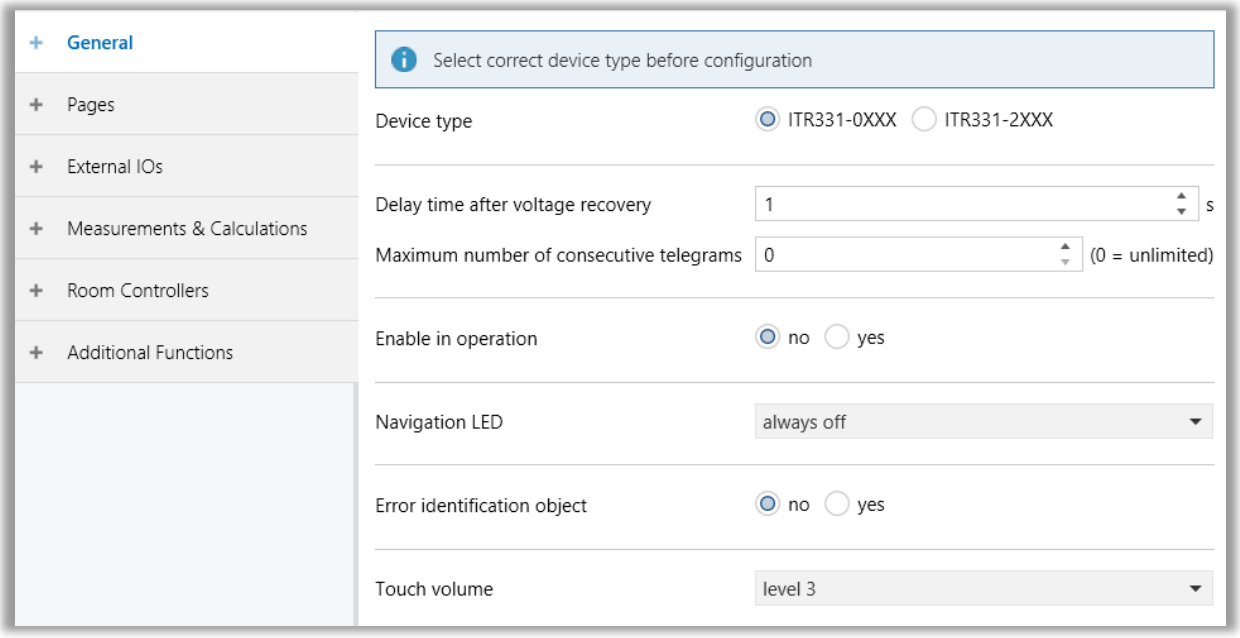


Fig. 4: General Parameter Configuration Page

According to the model of the device, the programmer can configure whether the LCD exists or not and the number of push buttons via corresponding tabs. To ensure that the models are selected correctly and also to be able to program correctly, the iX3 model appears on the screen as shown above.

4.1.A. Enable in Operation

This function has an important role to detect whether the device is working or not. By enabling the "Enable in operation" parameter, it is possible to know if the device is working properly. The value set in "in operation send" parameter is sent with a preset time via the "In Operation" object. If this telegram is received periodically, it shows that the device is working properly. Since the period time is in seconds, it is better to keep the period time higher in order not to increase the bus line traffic.

4.1.B. Error Identification

The faults which are sensor faults, digit overflow of the sensor value and out of the operation range of room controller etc., can be indicated via object.

Error Code	Cause
E0.1	Integrated temperature sensor fault
E0.2	Integrated humidity sensor fault
E0.3	Integrated air quality sensor fault
E0.5	Integrated brightness sensor fault
E0.6	External input - 1 sensor fault
E0.7	External input - 2 sensor fault

Table 4: Error Codes

Caution Code	Cause	Minimum Limit	Maximum Limit
C1.1	Integrated temperature out of range	-999	999
C1.2	Integrated humidity out of range	0	999
C1.3	Integrated air quality out of range	0	999
C1.5	Integrated brightness out of range	0	999
C1.6	External input - 1 sensor out of range	0 (brightness) -999 (temperature)	999
C1.7	External input - 2 sensor out of range	0 (brightness) -999 (temperature)	999
C1.8	Room temperature out of operational range	-60	60

Table 5: Caution Codes

➔ If any sensor fault is occurred, a string message is transmitted to KNX with error code. For example; if internal temperature sensor is on fault, "E0.1: True" message is sent. If error is fixed, "E0.1: False" message is sent.

➔ If room temperature is higher/lower than -60°C / 60°C, "E1.7: True" message is sent to indicate ambient temperature is too high/low.

➔ On measurement channel, internal and external sensor measurements are made. If sensor's output values are out of range as the table, related error code is sent over KNX bus.

4.1.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Device type	This parameter determines the device type.	**ITR331-0XXX** ITR331-2XXX
Delay time after voltage recovery (sec)	This parameter is used to determine the delay time after voltage recovery in seconds. When in a delayed state, the iX3 does not send any KNX telegrams. Incoming telegrams are received and updated in the background. The updated values are only executed when the wait state ends and then sent according to the parametrization.	**1**…60
Maximum number of consecutive telegrams **(0 = unlimited)**	This parameter is used to set the maximum number of sent telegrams by the device in the given time period.	**0**…255
-> Telegram period[1]	This parameter is used to determine the total period time of maximum number of consecutive telegrams. For example; "Maximum number of consecutive telegrams" is set 5 and "Telegram period" is set to 500ms. This means that maximum 5 telegrams can be sent along 500ms.	50 ms **100 ms** 200 ms 500 ms 1 s 2 s 3 s 5 s 10 s 30 s 1 min 2 min 3 min 4 min 5 min 10 min
Enable in operation	This parameter is used to determine the existence of the iX3 on the KNX bus line. The cyclic telegram can be monitored by an external KNX device. If a telegram is not received, the device may be defective or the KNX cable to the transmitting device may be interrupted. **No:** The group object is not enabled. **Yes:** The group object is enabled.	**No** Yes

-> In operation send[2]	This parameter is used to determine the send value of the "General - In operation" group object on the KNX bus line.	Alive value 0 **Alive value 1**
-> In operation send interval (min) [2]	This parameter is used to set the cyclically sending time interval value of the "General - In operation" group object.	1…**5**…255
Error identification object	This parameter is used to send an error message to indicate the error type if an error occurs. If it is selected "Yes", the "General – Error Identification" group object is visible.	**No** Yes
Touch volume	This parameter is used to determine the sound volume after touching the screen.	Disable Level 1 Level 2 **Level 3** Level 4 Level 5 Level 6 Level 7

[1] This parameter is visible when the function "Maximum number of consecutive telegrams" is set to "0".

[2] This parameter is visible when the function "Enable in operation" is set to "Yes".

4.1.2. Display Settings

The users can make many settings related to the LCD screen via ETS software. Many features such as screen theme, screen brightness, language selection, screen saver and password screen can be controlled with this tab.

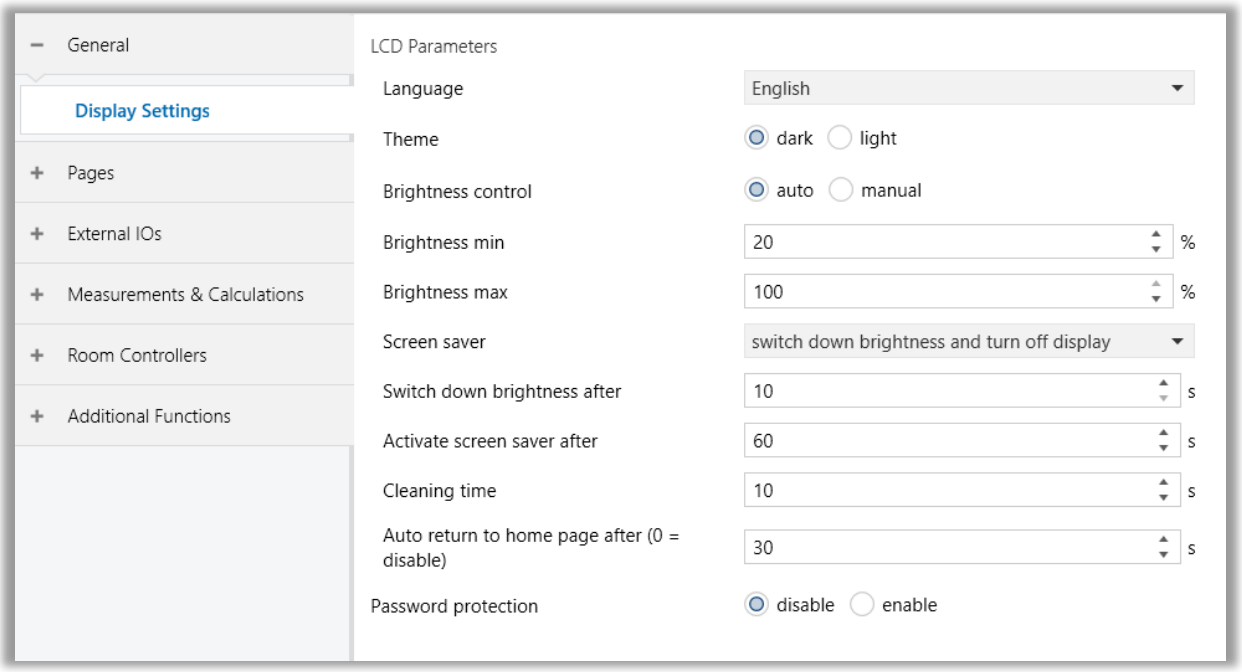


Fig. 5: Display Settings Configuration Page

4.1.2.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Language	This parameter determines the device language.	**English** Turkish German Russian Arabic French Greek Italian Persian Spanish Polish
Theme	This parameter determines the screen theme.	**Dark** Light
Brightness control	This parameter determines the brightness control type of the LCD.	**Auto** Manual
Brightness min	This parameter determines the minimum brightness of the LCD.	10…20…100 %
Brightness max	This parameter determines the maximum brightness of the LCD.	10…100 %
Screen saver	This parameter determines the type of screen saver that will be activated when the screen is not touched for a specified period of time.	Disable Turn off display Switch down brightness **Switch down brightness and turn off display** Switch down brightness and show digital clock Show digital clock
-> Turnoff display after[1]	The screen turns off after the time specified in this parameter.	10…**60**…255 s
-> Switch down brightness after[2]	The brightness of screen is dimmed to minimum brightness value after the time specified in this parameter.	10…**60**…255 s
-> Activate screen saver after[3]	The screen saver is activated after the time specified in this parameter.	10…60…255 s

Cleaning time	The cleaning screen is active for the time specified in the parameter.	1…**10**…255 s
Auto return to home page after	This parameter determines the delay time from the function page back to the home page when there is no operation on the device.	1…**30**…255 s
Password protection	This parameter determines whether to enable password function, that is, when entering screen saver or screen off, and whether to input a password when re-enter screen operation.	**Disable** Enable

[1] This parameter is visible when the function "Screen saver" is set to "Turn off display".

[2] This parameter is visible when the function "Screen saver" is set to "Switch down brightness" or "Switch down brightness and turn off display" or "Switch down brightness and show digital clock".

[3] This parameter is visible when the function "Screen saver" is set to "Switch down brightness and turn off display" or or "Switch down brightness and show digital clock" or "Show digital clock".

4.1.3. Password Settings

It is mainly setting password function and the object type of output value sent to the bus after device is activated through password.

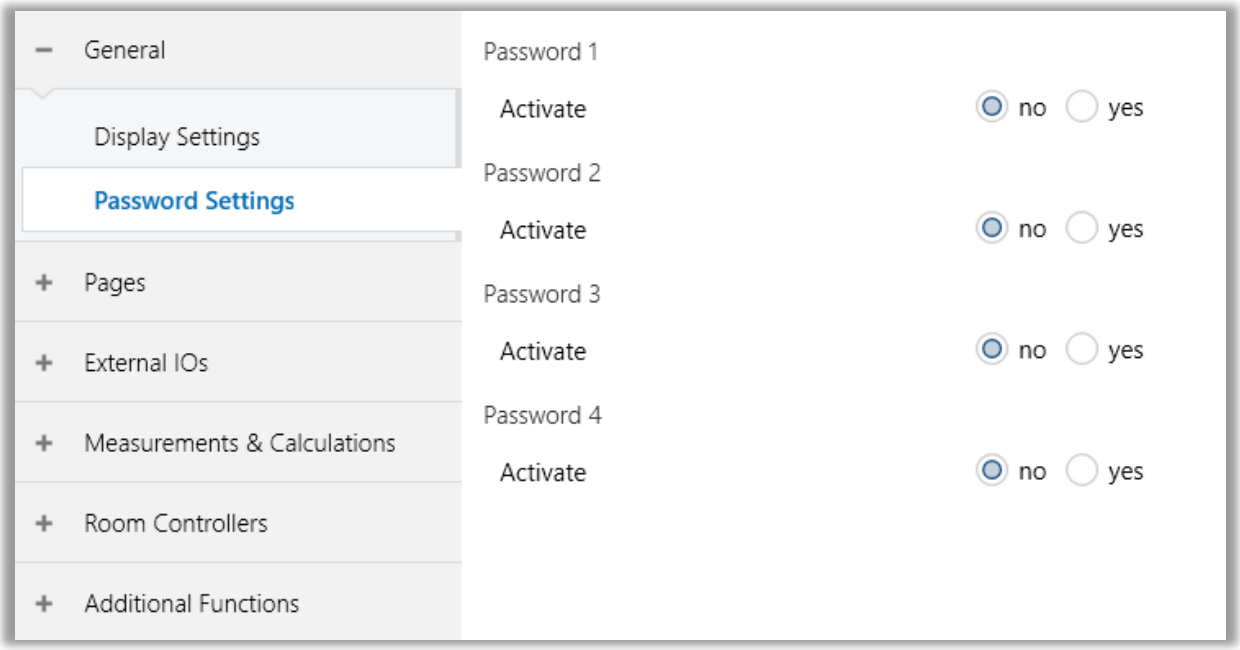


Fig. 6: Password Settings Configuration Page

4.1.3.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Password X Activate	This parameter is used to active the password.	**No** Yes
->Reaction for password[1]	This parameter determines the object type of output value sent to the bus after the device is activated through a password.	**None** Bit Byte Percentage Scene
->Output value[2]	This parameter determines the output value sent to the bus when the password is triggered, the range of output value is determined by the selected data type.	Values depend on DPT selection.
->Delay time for sending output[2]	This parameter determines the delay time for sending.	**0**…255 s

[1] This parameter is visible when the function "Activate" is set to "Yes".

[2] This parameter is visible when the function "Reaction for password" is **not** set to "None"

4.2. Pages

4.2.1. Home Page

The users can configure the layout and page navigation function of the home page, up to 2 home pages can be set, and up to 8 pages in the Multifunction page can be set. The parameters for each navigation are the same and can be linked to the specified function page.

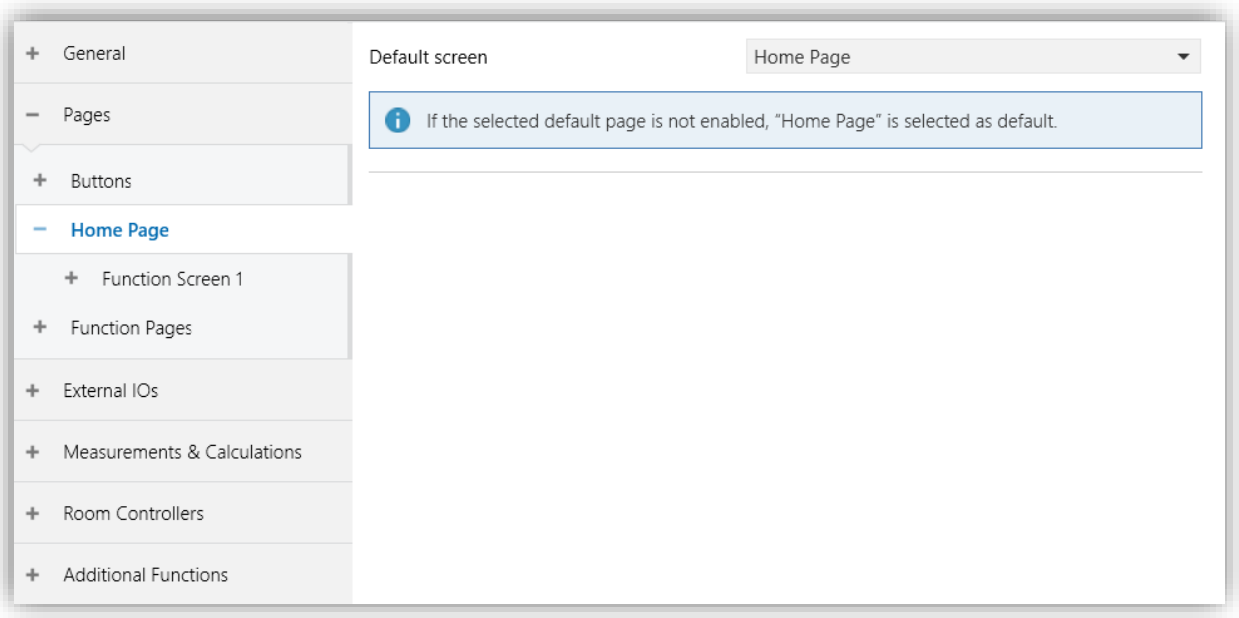


Fig. 7: Home Page Configuration Page

4.2.1.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Home page screen count	This parameter determines the page count of home screen.	1...**2**…5
-> X. screen	This parameter determines the X. page of home screen. If a screen is selected more than 1, the second screen is ignored.	Function Screen 1 Function Screen 2 Weather Forecast Screen Sensor Information Screen Meter Information Screen
Default screen	This parameter determines the default page of the home screen. The page specified in the parameter is shown first upon returning to the home page after the specified time in the parameter. If the selected default screen is not in the screen list, the selected parameter is ignored and the default screen is assigned as 1. screen.	**Function Screen 1** Function Screen 2 Weather Forecast Screen Sensor Information Screen Meter Information Screen

4.2.2. Function Screen X

The users can configure the functions page's count and layout. Up to 6 different layouts are available in this section. Up to 8 function pages can be set.

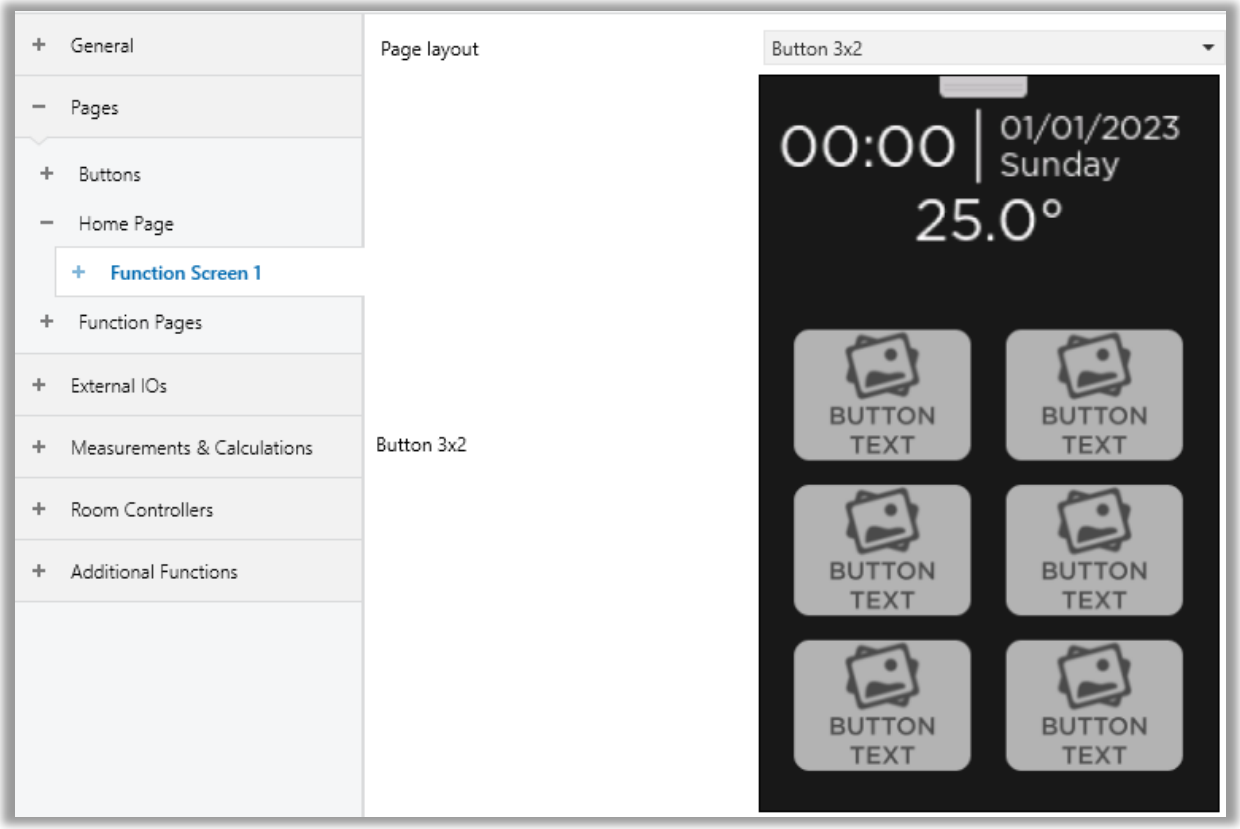


Fig. 8: Default Function Page Configuration Page

4.2.2.1 Parameters List

PARAMETERS	DESCRIPTION	VALUES
Page layout	This parameter determines the page's layout.	Button 1 x 1 Button 1 x 2 Button 2 x 1 Button 2 x 2 Button 3 x 1 **Button 3 x 2**

4.2.3. Home Page - Button X

Each button is linked to different function such as switch function, multifunction, thermostat control, AC control functions.

The users can configure the button's colour, icon and text. Up to 170 icons are available. Up to 22 characters can be input into the button's name. The feedback bar can be added bottom of the button to indicate feedback value according to function type.

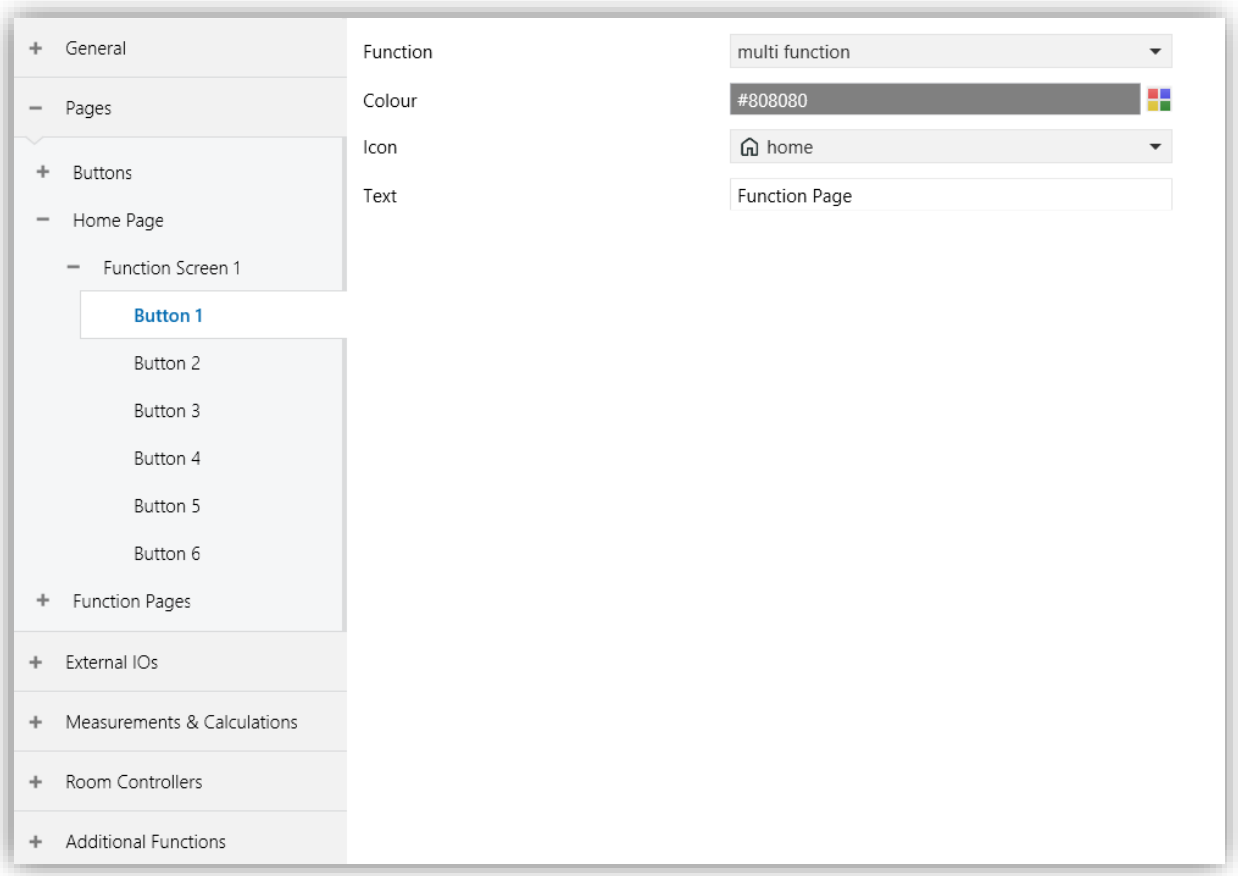


Fig. 9: Home Page - Button X Configuration Page

4.2.3.1 Parameters List

PARAMETERS	DESCRIPTION	VALUES
Function	This parameter determines the function of the page button.	None Dummy **Multifunction** Switch function Thermostat control Air conditioner control
-> Button function[1]	This parameter determines the switch function type. ***See "Switch Functions" section for detail information.***	No function Switch Switch/dimming Shutter/blinds Value / forced operation Scene control Mode selection Command sequence Counter RGB colour control RGBW control Thermostat extension Colour control Music control
-> Connect to[2]	This parameter determines the channel number to which the thermostat or AC control is connected. If the button's function is selected as thermostat control or air conditioner control, the button navigates to a thermostat or AC control screen according to selection.	**Channel 1** Channel 2 Channel 3 Channel 4
Colour	This parameter determines the colour of the page button.	0x000000… **0x808080**…0xFFFFFF
Icon	This parameter determines the icon of the page button.	Values depend on selection
Text	This parameter determines the text of the page button.	Up to 22 characters (UTF-8)
Feedback	This parameter determines the feedback type of the page button.	**None** On/Off Percentage

[1] This parameter is visible when the function "Function" is set to "Switch function".
[2] This parameter is visible when the function "Function" is set to "Thermostat control" or "Air conditioner control".

4.2.4. Function Pages

Functions pages can be configured if a button function is selected as Multifunction. In multifunction page, up to 8 buttons can be set. Each button has different button function the same as Function Screen. The users can set up to 16 function pages.

4.2.5. Page X

The user can configure each function page in this section. Different button layouts are available for function pages. Up to 8 buttons can be set and up to 5 different button layouts can be set.

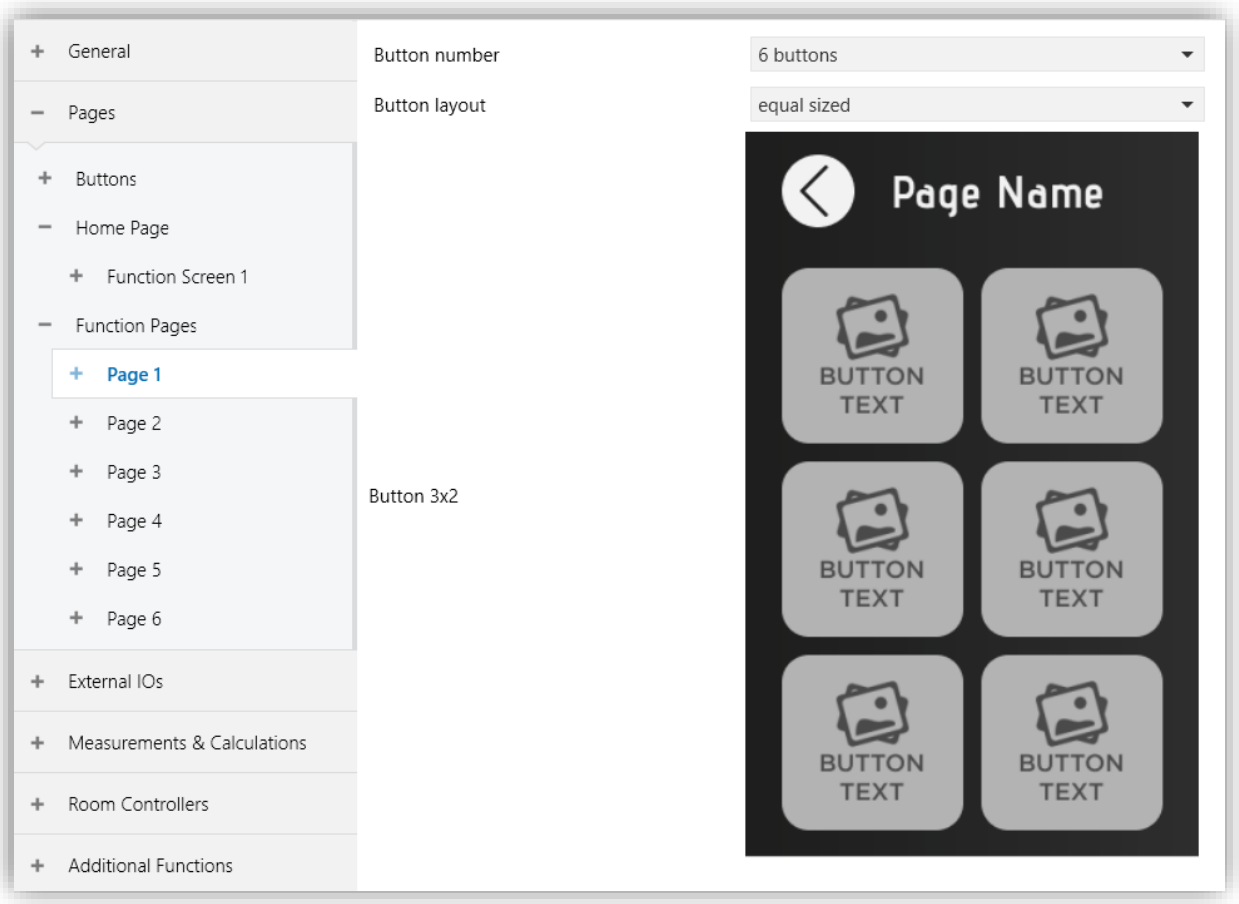


Fig. 10: Page X - Configuration Page

4.2.5.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Button number	This parameter determines the button number of function pages.	2 buttons 4 buttons 6 buttons
Button layout	This parameter determines the button layout of function pages.	Equal sized Long short short long Short long long short Left long right short Left short right long

4.2.6. Function Page - Button X

Each button is linked to different function such as switch function, colour control, music control, thermostat control, AC control functions.

The users can configure the button's colour, icon and text. Up to 170 icons are available. Up to 22 characters can be input into the button's name. The feedback bar can be added bottom of the button to indicate feedback value according to function type.

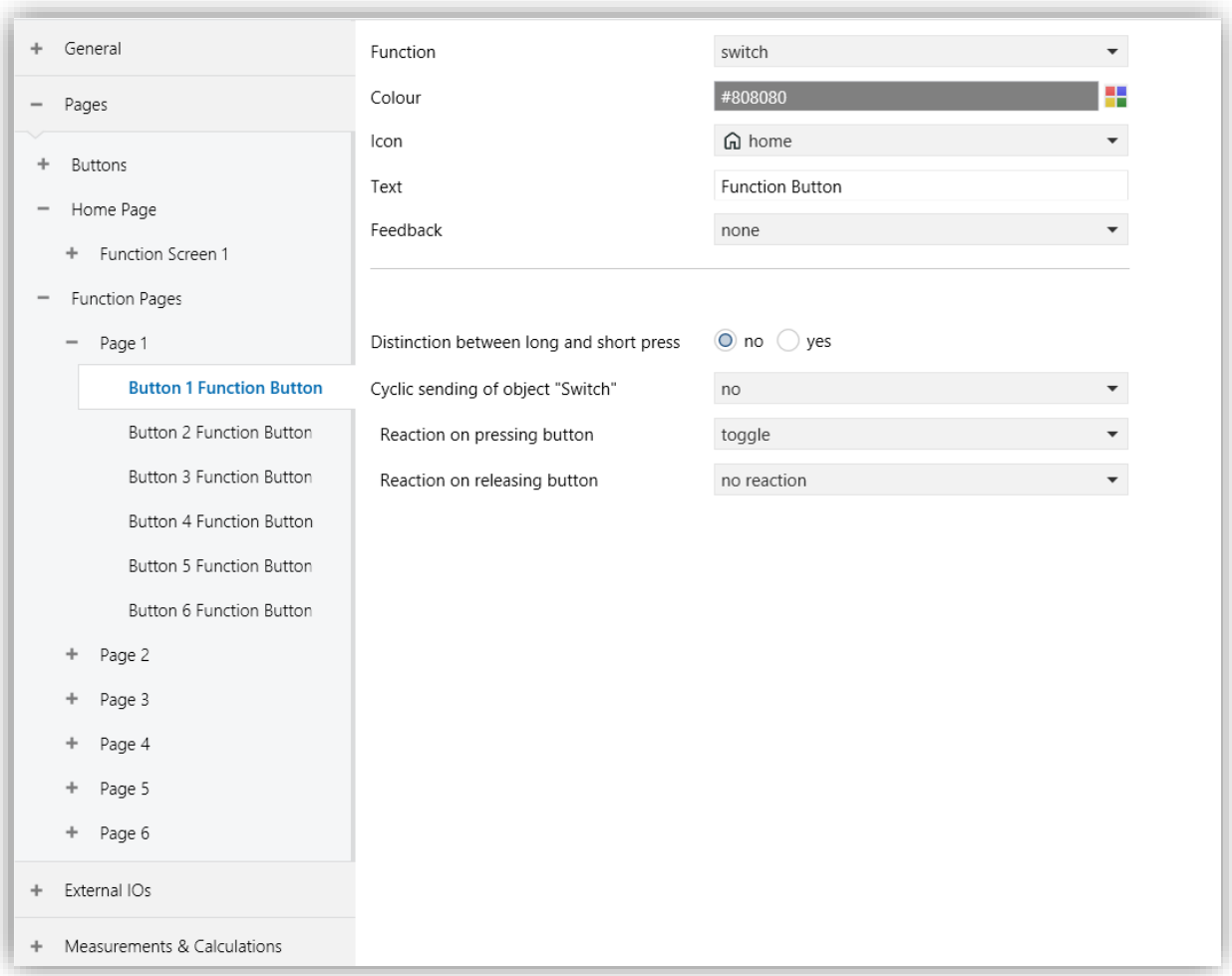


Fig. 11: Function Page - Button X Configuration Page

4.2.5.1 Parameters List

PARAMETERS	DESCRIPTION	VALUES
Function	This parameter determines the switch function type. If the function is selected as thermostat control or air conditioner control, function button navigates thermostat or air conditioner screen. ***See "Switch Functions" section for detail information.***	No function **Switch** Switch/dimming Shutter/blinds Value / forced operation Scene control Mode selection Command sequence Counter RGB colour control RGBW control Thermostat extension Colour control Music control Thermostat control Air conditioner control
-> Connect to[1]	This parameter determines the channel number to which the thermostat or AC control is connected. If the button's function is selected as thermostat control or air conditioner control, the button navigates to a thermostat or AC control screen according to selection.	**Channel 1** Channel 2 Channel 3 Channel 4
Colour	This parameter determines the colour of the page button.	0x000000… **0x808080**…0xFFFFFF
Icon	This parameter determines the icon of the page button.	Values depend on selection
Text	This parameter determines the text of the page button.	Up to 22 characters (UTF-8)
Feedback	This parameter determines the feedback type of the page button.	**None** On/Off Percentage

[1] This parameter is visible when the function "Function" is set to "Thermostat control" or "Air conditioner control".

4.2. Switch Functions

4.2.1. Switching

This function is used to perform the switching operation. Depending on the settings configured in the switching process, when the button is pressed or released, the ON or OFF values are generated. After each operation, a telegram is sent to the KNX bus line. Telegram is generated based on the configured settings.

If you want to configure the push button with the "switching" function, choose it from the parameter page and then a new object will appear under the device object list on the left side. This object's name is "switching". General configurations are made via this object. When the "switching" function is enabled, it is added to the object list of the device. After assigning the group address to this object, attention should be paid to the type of data it uses. It is a good technique to use default data types.

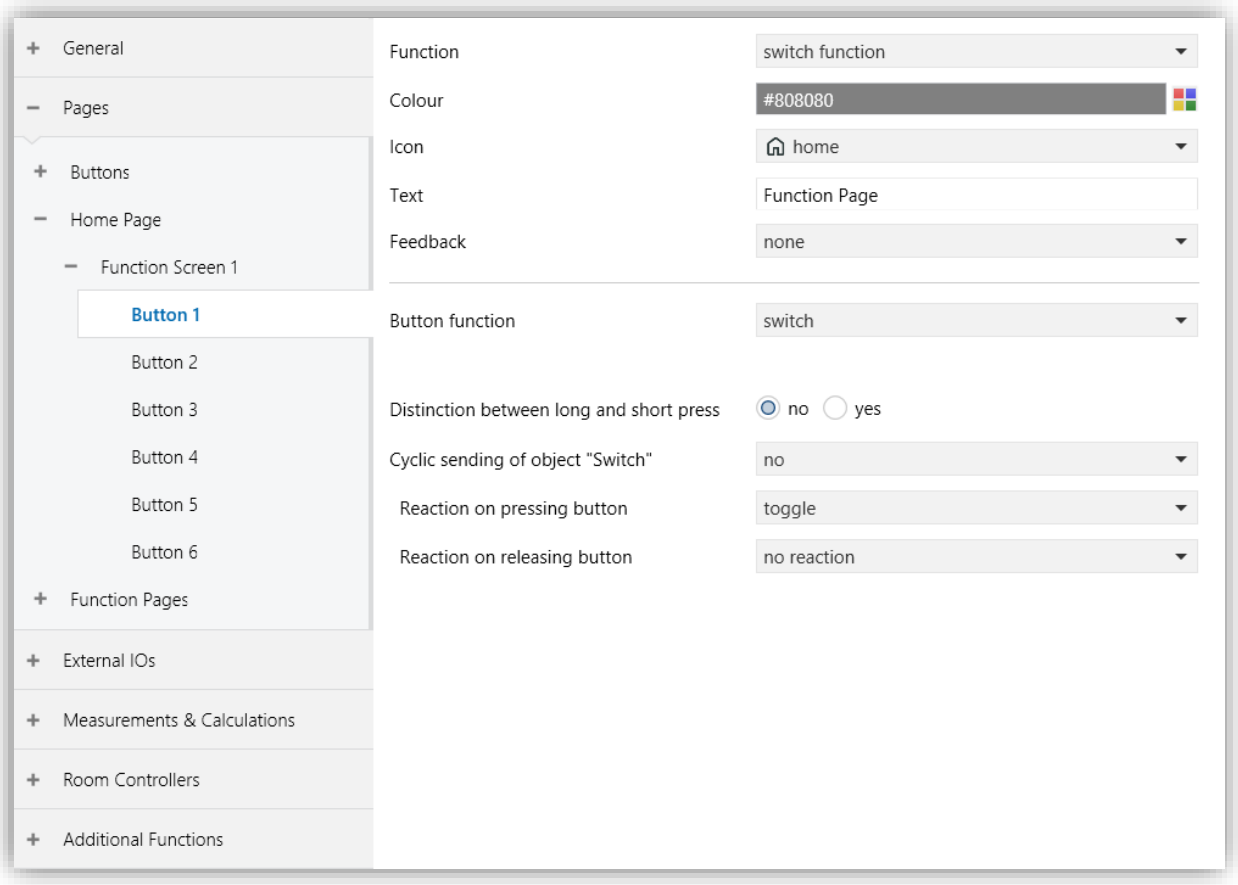


Fig. 12: Switching Function Configuration

4.2.1.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Button name	This parameter is used to type a button name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Button function	This parameter is used to determine the button function. If no function is selected, Button X will not be used. For other choices, all functionalities are configured separately.	No function **Switch** Switch/dimming Shutter/blinds Value / forced operation Scene control Mode selection Command sequence Counter RGB colour control RGBW control Thermostat Extension
Distinction between long and short press: No		
Cyclic sending of object "Switch"	This parameter is used to periodically send the commands to the bus line.	**No** If "Switch" = ON If "Switch" = OFF Always
-> Telegram repeated every[1]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams	00:00:01…**00:08:20**…18:12:15
Reaction on pressing button	This parameter is visible if there is a distinction between pressing and releasing operations. It is used to determine the pressing operation sending the value of the Button X.	No reaction On Off **Toggle**
Reaction on releasing button	This parameter is visible if there is a distinction between pressing and releasing operations. It is used to determine the releasing operation sending the value of the Button X.	**No reaction** On Off Toggle

Send button value after bus voltage recovery	This parameter is used to determine the sending value of the inputs when the bus voltage has been recovered.	**No** Yes
Distinction between long and short press: Yes		
Cyclic sending of object "Switch"	This parameter is used to periodically send the commands to the bus line.	**No** If "Switch" = ON If "Switch" = OFF Always
Reaction on short press	This parameter is visible if there is a distinction between short and long operations. It is used to determine the short press operation sending the value of the Button X.	**No reaction** On Off Toggle
Reaction on long press	This parameter is visible if there is a distinction between short and long operations. It is used to determine the long press operation sending the value of the Button X.	**No reaction** On Off Toggle
Long press after	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	00:00.200…**00:00.500** …01:05.535
Number of object for short/long press	This parameter is used to determine the object count to use for short and long operations. **1 object:** short and long operations will proceed with the same object. **2 objects:** short and long operations will proceed with 2 different objects.	**1 object** 2 objects

[1] This parameter is visible when the parameter "Cyclic sending of object "Switch" is set to "If "Switch" = ON" or "If "Switch" = OFF" or "Always".

4.2.2. Switch / Dimming

This feature enables increasing or decreasing of lighting circuit's lighting level. There is 2 functionality such as "only dimming" and "dimming and switching". Also, each functionality has 2 dimming mode such as "start/stop dimming" and "step dimming".

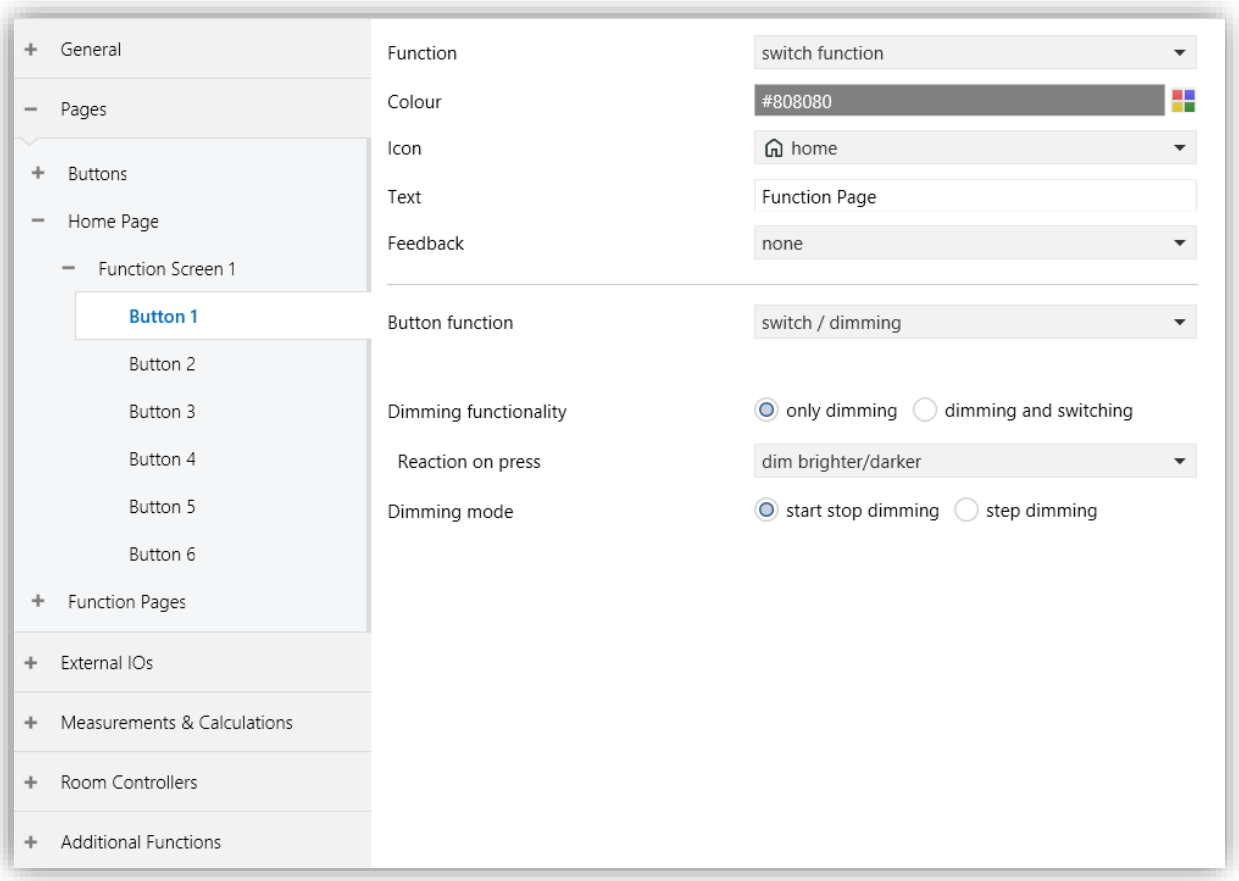


Fig. 13: Switch/Dimming Function Configuration

If the "Only dimming" function is enabled, dimming control is done via only a group object on press operation. If the "Dimming and switching" function is enabled, also another group object is available for switching function on short press operation and another group object is available for dimming function on long press operation.
In start/stop dimming mode, if the button is pressed, the dimming value (4-bit) is sent via the "dimming" object. If the button is released, the "stop" telegram is sent to the bus line and dimming control is over.
In step dimming mode, if the button is pressed, the dimming value (4-bit) is sent step by step via the "dimming" object. The step value is determined via the "Brightness change on every sent" parameter. Each step is sent cyclically according to "Sending cycle time: Telegram is repeated every" parameter.

If "Reaction on press" parameter is selected as "Dimming absolute", "Page" and "Slider" percentage control types are enabled. "Page" control type is navigated to "Dimming control page", "Slider" control type is shown a slider effect on the button to control value. Short press in the "Slider mode", "Stop" telegram is sent to the bus line and dimming control is over.

4.2.2.1. Parameters List

PARAMETERS	DESCRIPTIONS	VALUES
Button name	This parameter is used to type an input name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Button function	This parameter is used to determine the input x operation mode. If no function is selected, Button X will not be used. For other choices, all functionalities are configured separately.	No function **Switch** Switch/dimming Shutter/blinds Value/forced operation Scene control Mode selection Command sequence Counter RGB colour control RGBW control Thermostat Extension
Dimming functionality	This parameter is used to define if the lighting can only be dimmed "Only dimming" or if additional switching is also permitted "Dimming and switching". In this case, a long button presses dims and a short button pushes switches.	**Only dimming** Dimming and switching
-> Reaction on press[1]	A distinction is not made between short and long operations here. It is used to determine the press operation sending the value of the Button X.	Dim brighter Dim darker **Dimming brighter/darker** Dim absolute
-> Reaction on short press[3]	This parameter is visible if there is a distinction between short and long operations. It is used to determine the short press operation sending the value of the Button X.	**No reaction** On Off Toggle
-> Reaction on long press[3]	This parameter is visible if there is a distinction between short and long operations. It is used to determine the long press operation sending the value of the Button X.	Dim brighter Dim darker **Dimming brighter/darker** Dim absolute
-> Percentage control type[2]	If reaction on press/reaction on long press parameter is selected as "Dim absolute", percentage dimming control is available. **Page:** Navigate to dimming control page. **Slider:** Control dimming via slider on the button.	**Page** Slider

-> Dimming direction after switch ON[4]	This parameter is used to determine the dimming direction when the switch object is "ON" on long operation.	**Brighter** Darker
-> Long press after[2]	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	00:00.200…**00:00.500** … 01:05.535
Dimming mode	This parameter is used to determine the dimming mode. Normal "Start-stop-dimming" starts the dimming process with a telegram BRIGHTER or DARKER and ends the dimming process with a STOP telegram. Cyclic sending of the telegram is not necessary in this case. With "Step dimming", the dimming telegram is sent cyclically during a long operation. The STOP telegram ends the dimming process at the end of the operation.	**Start stop dimming** Step dimming
-> Brightness change on every sent telegram[5]	This parameter is set to change the brightness (in per cent), which is cyclically sent with every dimming telegram.	100% 50% **25%** 12.5% 6.25% 3.125% 1.563%
-> Sending cycle time: Telegram is repeated every[5]	This parameter is used to determine the sending cycle time. The dimming telegram is sent cyclically during a long operation if "Dimming steps" are set. The cycle time for sending corresponds with the time interval between two telegrams during cyclical sending.	0.3s 0.4s **0.5s** 0.6s 0.8s 1s 1.2s 1.5s 2s 3s 4s 5s 6s 7s 8s 9s 10s

[1] This parameter is visible when the parameter "Dimming functionality" is set to "Only dimming".
[2] This parameter is visible when the parameter "Reaction on press" is set to "Dimming absolute".
[3] This parameter is visible when the parameter "Dimming functionality" is set to "Dimming and switching".
[4] This parameter is visible when the parameter "Reaction on long press" is set to "dimming brighter/darker".
[5] This parameter is visible when the parameter "Dimming mode" is set to "Step dimming".

4.2.3. Shutter/Blinds

A shutter/blinds circuit can be controlled up–down or on-off methods with a "slat angle/stop" object courtesy of this feature. Each function has 2 different "up / down" and "slat angle/stop" objects. At the control of the shutter/blinds circuit, a short press of the button sends a "step movement" telegram and a long press of the button sends a "nonstop movement" telegram to the bus line. A shutter/blinds circuit is controlled by "1 button toggle" or "2 buttons up/down" control modes.

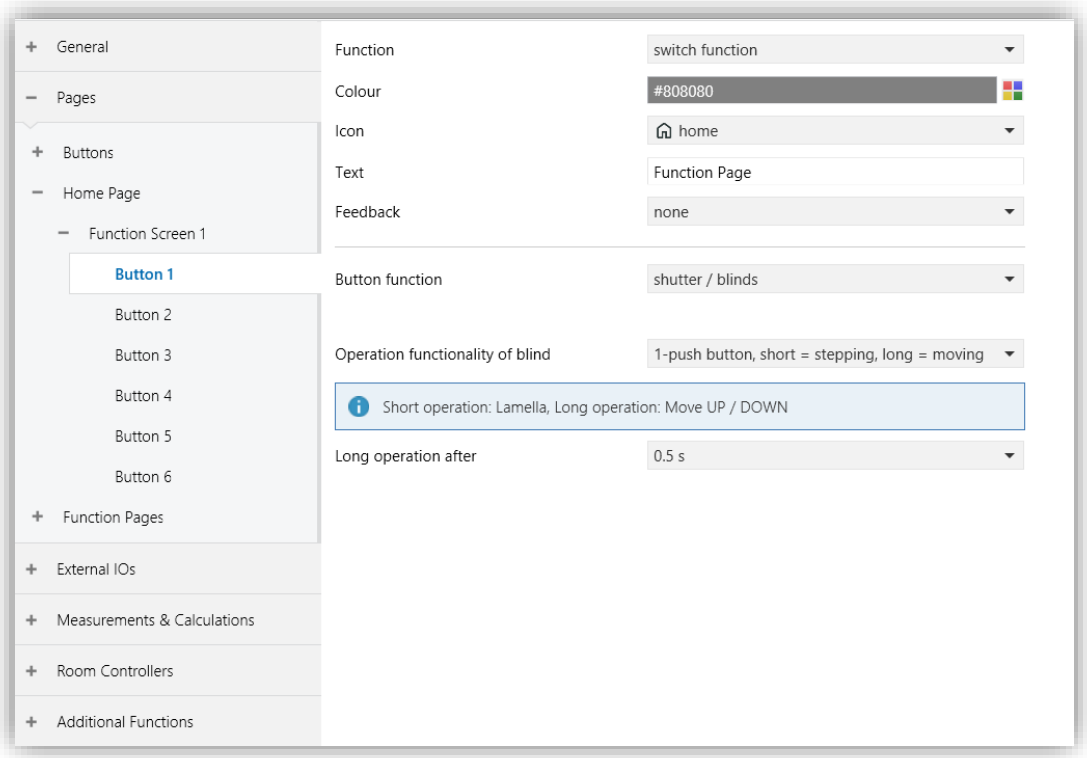


Fig. 14: Shutter/Blinds Function Configuration

Shutter/blinds circuit control with 1 button; Push up, pull down and stop controls can be done with 1 push button. At every time of short press, the push button will send the following sequential values in the form of; down movement, stop, up movement and stop. The movement aspect of the shutter or slat angle adjustment aspect always depends on the previous action. There is a push-button status object to prevent sending wrong commands to the bus line and the current values of the object can be updated by the devices at the same KNX bus line. This object must be connected to the actuator's status parameter via a related group address.

Shutter/blinds circuit control with 2 buttons; 2 buttons must be used for this option. If both buttons are configured, with long press action the shutter can be moved up or down and with short press action, the movement stops or slat angle step movement can be configured. The minimum time to detect the long press action is configured via a parameter. Every command controls the buttons defined as "Up" or "Down" via the "Direction" parameter. When short pressed to the button configured as "up", it sends an "up" value to the bus line, and when short pressed to the button configured as "down", it sends a "down" value to the bus line.

Shutter/blinds circuit control with 1 button and page/slider; Push up, pull down and stop controls can be done with 1 push button. At every time of short press, the push button will send the following sequential values in the form of; down movement, stop, up movement and stop. The movement aspect of the shutter or slat angle adjustment aspect always depends on the previous action. There is a push-button status object to prevent sending wrong commands to the bus line and the current values of the object can be updated by the devices at the same KNX bus line. This object must be connected to the actuator's status parameter via a related group address.

If "operation functionality of blind" parameter is selected one that long operation action is "Percentage" or "Only percentage", "Control Type" parameter is enabled. 4 different control types are available: Curtain, Blind, Jalousie, Percentage bar. Each one has different control types. Curtain, Blind and Jalousie are navigated to control page for percentage value. "Percentage bar" control type is shown a slider effect on the button to control value. Short press in the "Slider mode", "Stop movement" telegram is sent to the bus line and position control is over.

4.2.3.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Button name	This parameter is used to type an input name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Button function	This parameter is used to determine the input x operation mode. If no function is selected, the input x will not be used. For other choices, all functionalities are configured separately.	No function Switch Switch/dimming **Shutter/blinds** Value/forced operation Scene control Mode selection Command sequence Counter RGB colour control RGBW control Thermostat Extension
Operating functionality of blind	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	1-push button, short = stepping, long = moving[1], 1-push button, short = moving, long = stepping[2], 1-push button operation[3], 1-switch button operation[4], 2-push button, standard[5] 2-switch operation, moving[6], 2-push button operation, moving[7], 2-push button operation, stepping[8], 1-push button, short = stepping, long = percentage[9], 1-push button, short = moving, long = percentage[10], Only percentage[11]

1-push button, short = stepping, long = moving		
Long operation after	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	0,3s, 0,4s, **0.5s**, 0.6s, 0.8s, 1.0s, 1.2s, 1.5s, 2s, 3s, 4s, 5s, 6s, 7s, 8s, 9s, 10s
1-push button, short = moving, long = stepping		
Long operation after	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	0,3s, 0,4s, **0.5s**, 0.6s, 0.8s, 1.0s, 1.2s, 1.5s, 2s, 3s, 4s, 5s, 6s, 7s, 8s, 9s, 10s
"STOP/lamella adj," is repeated every	This parameter is used to determine the time between two telegrams is set. This parameter is visible in operations in which the object "STOP/lamella adjustment" is sent cyclically on the bus during a long operation.	0,3s, 0,4s, **0.5s**, 0.6s, 0.8s, 1.0s, 1.2s, 1.5s, 2s, 3s, 4s, 5s, 6s, 7s, 8s, 9s, 10s
2-push button operation, standard		
Reaction on short operation	This parameter is used to determine the reaction when an operation occurs. A distinction is not made between short and long operations here.	**Stop/lamella up** Stop/lamella down
Reaction on long operation	This parameter is visible if there is a distinction between short and long operations. It is used to determine the long press operation sending the value of the Button X.	**Move up** Move down
Long operation after	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	0,3s, 0,4s, **0.5s**, 0.6s, 0.8s, 1.0s, 1.2s, 1.5s, 2s, 3s, 4s, 5s, 6s, 7s, 8s, 9s, 10s
2-switch operation, moving		
Reaction on press	It is used to determine the press operation sending the value of the Button X.	**Move up** Move down
2-push button operation, moving		
Reaction on press	It is used to determine the press operation sending the value of the Button X.	**Move up** Move down
2-push button operation, stepping		
Reaction on press	It is used to determine the press operation sending the value of the Button X.	**Stop/lamella up** Stop/lamella down
1-push button, short = stepping, long = percentage		
Control Type	It is used to determine the page type to be opened.	**Curtain** Roller Jalousie

		Percentage Bar
Long operation after	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	0,3s, 0,4s, **0.5s**, 0.6s, 0.8s, 1.0s, 1.2s, 1.5s, 2s, 3s, 4s, 5s, 6s, 7s, 8s, 9s, 10s
1-push button, short = moving, long = percentage		
Control Type	It is used to determine the page type to be opened.	**Curtain** Roller Jalousie Percentage Bar
Long operation after	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	0,3s, 0,4s, **0.5s**, 0.6s, 0.8s, 1.0s, 1.2s, 1.5s, 2s, 3s, 4s, 5s, 6s, 7s, 8s, 9s, 10s
Only percentage		
Control Type	It is used to determine the page type to be opened.	**Curtain** Roller Jalousie Percentage Bar

[1] Short operation: Lamella, Long operation: Move UP / DOWN
[2] Short operation: Move UP/DOWN, Long operation: Lamella
[3] On every operation in succession: UP – DOWN – STOP
[4] On operation: UP / DOWN, End of operation: STOP
[5] Short operation: STOP – Lamella UP / DOWN, Long operation: Move UP / DOWN
[6] On operation: Moving End of operation: STOP
[7] On operation: Moving
[8] On operation: Stepping
[9] Short operation: Lamella, Long operation: Navigate the page specified in "Control Type" parameter.
[10] Short operation: Move UP/DOWN, Long operation: Navigate the page specified in "Control Type" parameter
[11] Long operation: Navigate the page specified in "Control Type" parameter

4.2.4. Value/Forced Operation

In this section, it is explained how to control an automation unit via iX3 via a value/forced via buttons connected to digital inputs. Detailed information on the relevant parameter configurations is described in the table below.

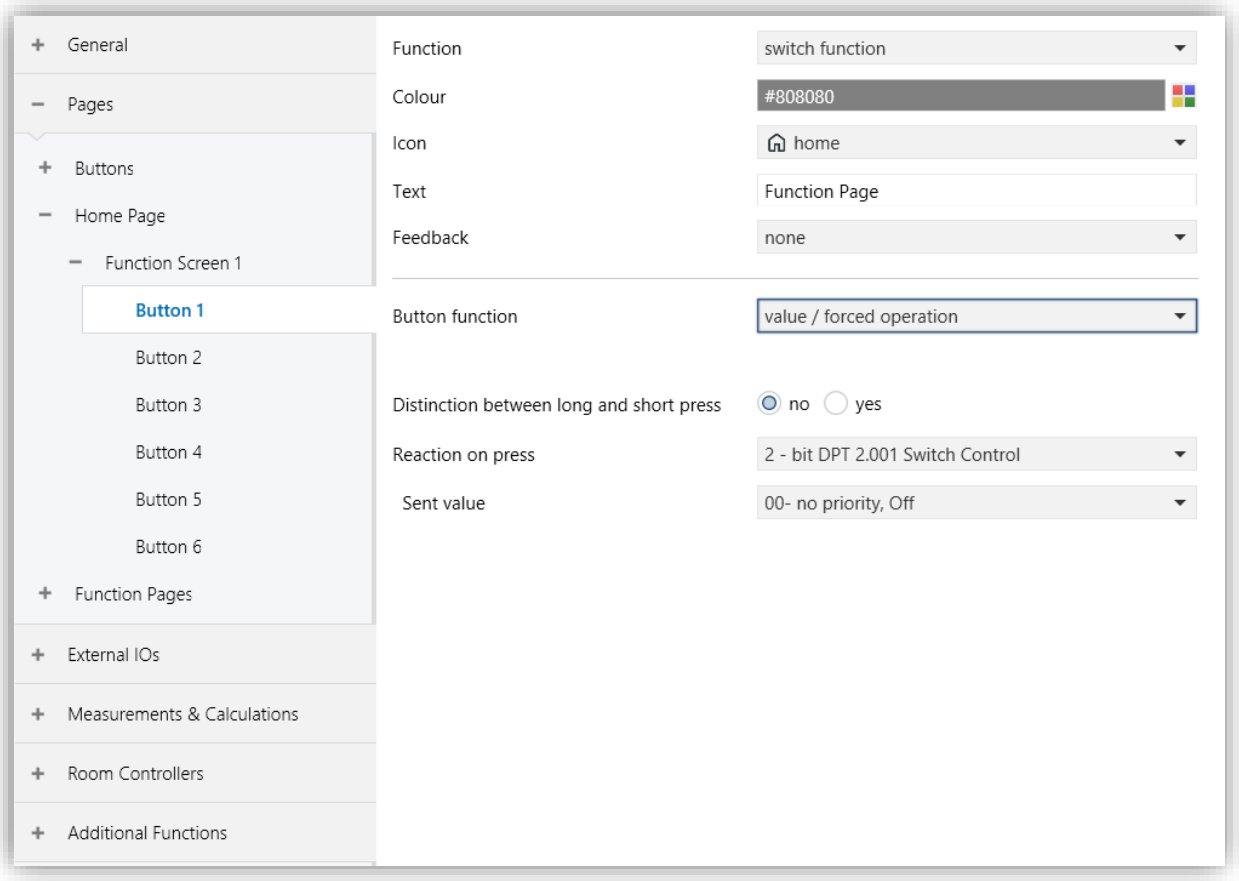


Fig. 15: Value/Forced Operation Function Configuration

4.2.4.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Button name	This parameter is used to type an input name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Button function	This parameter is used to determine the Button X operation mode. If no function is selected, Button X will not be used. For other choices, all functionalities are configured separately.	No function Switch Switch/dimming Shutter/blinds **Value/forced operation** Scene control Mode selection Command sequence Counter RGB colour control RGBW control Thermostat Extension
Distinction between long and short press	This parameter is used to set if the input differentiates between short and long operations. With the option "yes", after opening/closing of the contract, it must, first of all, be ascertained if a short or long operation has occurred here. Only thereafter will a possible reaction be triggered.	**No** Yes
-> Long press after[1]	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	00:00.200… **00:00.500** … 01:05.535
-> Reaction on long press[1]	This parameter is visible if there is a distinction between short and long operations. It is used to determine the long press operation sending the value of the Button X.	**2-bit DPT 2.001 Switch Control** 1 Byte DPT 5.001 Percent (0…100%) 1 Byte DPT 5.005 Decimal factor (0…255) 1 Byte DPT 17.001 Scene number

		2 Byte DPT 7.600 Colour temperature (Kelvin) 2 Byte DPT 9.001 Temperature (°C) 2 Byte DPT 9.004 Brightness (lux) 3-Byte DPT 232.600 RGB value 3x (0...255)
-> Sent Value[1]	This parameter is used to determine the sending value to the bus when a long operation occurs.	Values depend on DPT selection.
Reaction on press	This parameter is visible if there is a distinction between short and long operations. It is used to determine the short press operation sending the value of the Button X.	**2 – bit DPT 2.001 Switch Control** 1Byte DPT 5.001 Percent (0…100%) 1Byte DPT 5.005 Decimal factor (0…255) 1Byte DPT 17.001 Scene number 2Byte DPT 7.600 Colour temperature (Kelvin) 2Byte DPT 9.001 Temperature (°C) 2Byte DPT 9.004 Brightness (lux) 3-Byte DPT 232.600 RGB value 3x (0…255)
Sent Value	This parameter is used to determine the sending value to the bus when a short operation occurs.	Values depend on DPT selection.

[1] This parameter is visible when the parameter "Distinction between long and short press" is set to "Yes".

4.2.5. Scene Control

The scene function is used to control devices and make pre-registration of their status with the push button which sends a command via a related group address. This feature allows one to register a setting as a scene and after a while, when the same settings or conditions are requested, each device can be activated only with 1 command instead of configuring them separately.

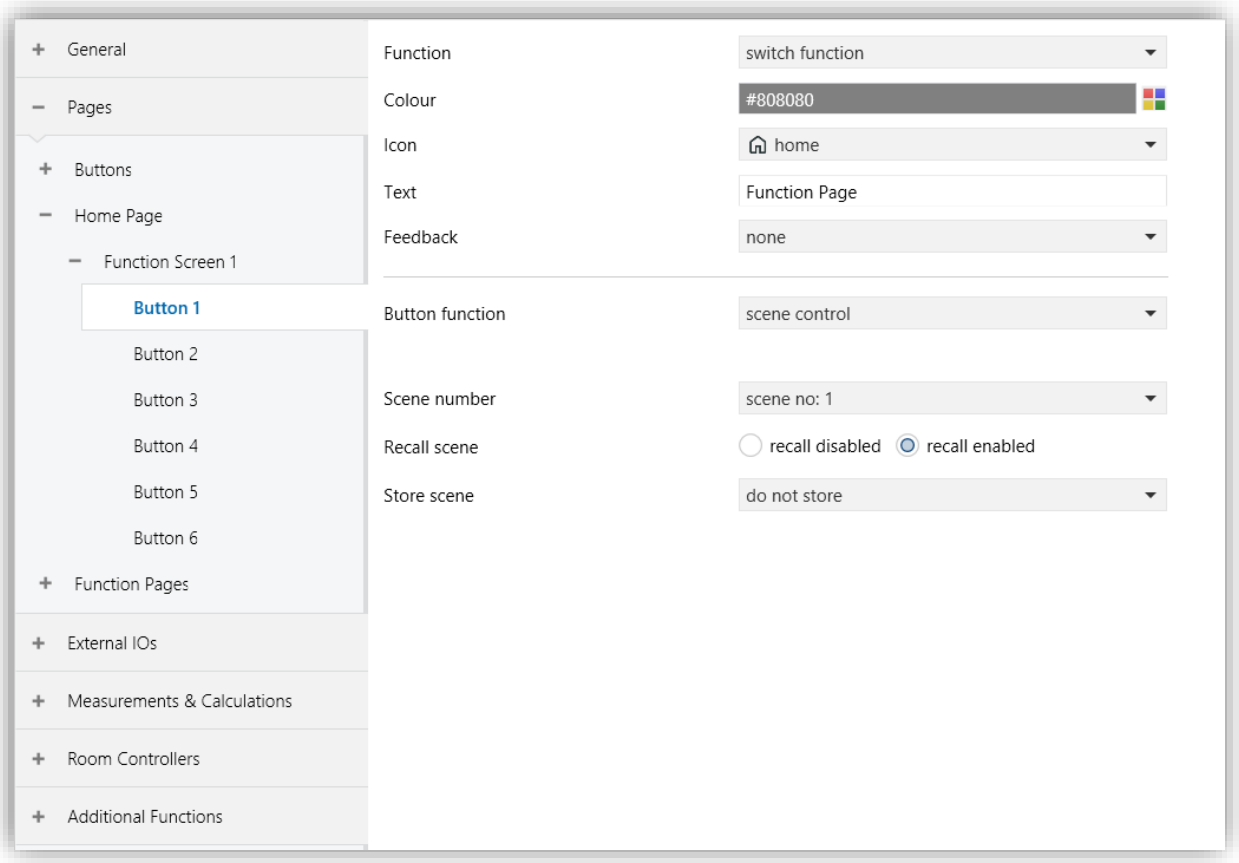


Fig. 16: Scene Control Function Configuration

This feature in the button sends telegrams that contain "scene run" or "scene register" functions, via the "scene" object. Scene numbers between 1 and 64 can be selected via the related group address. The scene number configured in the button must match the scene number configured on the parameters in other devices. Scene number (1 – 64) is used to run the scene using the related object. The values sent via related object must be as in the form "Scene Number + 128" for storing the scene feature.

If a scenario number is configured as 2 and it is wished to register this scenario, a value of 130 should be sent (128 + 2). If the scenario number is configured as 24, the value of 152 (128 + 24) should be sent for the scenario registering feature.

To run every scene, a time-delayed is defined or not in the parameters should be checked, whether to send with or without time delay. This feature allows the creation of dynamic scene arrays in which several outputs connect with time delay.

After programming with ETS, scene values that are used for parameterization will be written to the actuator. This means related scenes will be erased and defined by the customer. Hence, before any maintenance, all configurations should be gotten by the programmer and whether the customer wants to use the same conditions.

4.2.5.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Button Name	This parameter is used to type an input name. The name can be consisting of 40 characters	**40 Bytes allowed**
Button Function	This parameter is used to determine the Button X operation mode. If no function is selected, Button X will not be used. For other choices, all functionalities are configured separately.	No function Switch Switch/dimming Shutter/blinds Value/forced operation **Scene control** Mode selection Command sequence Counter RGB colour control RGBW control Thermostat Extension
Scene number	This parameter is used to give the scenario number to the generated scenario before.	**Scene no: 1** … 64
Recall scene	This parameter is used to determine the recall of the scene. If this parameter is selected as "recall enabled" the configured scene number will be called.	Recall disabled **Recall enabled**
Store scene	This parameter is used to determine whether to store or not store the related scene. **On long operation:** The scene will be stored after a long operation. **With "Store scene" obj. value = 1:** The scene will be stored on operation if the Store scene object value is 1. **On long operation ("Store scene" obj. value = 1):** The scene will be stored on long operation if the Store scene object is 1.	**Do not store** On long operation With "store scene" obj value = 1 On long operation ("store scene" obj value = 1)
-> Long press after[1]	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	00:00.200… **00:00.500** … 01:05.535

[1] This parameter is visible when the parameter "Long press after" is set to "On long operation" or "On long operation ("store scene" obj value = 1)".

4.2.6. Mode Selection

This section, it is explained how to control the operating modes of an HVAC unit via the buttons connected to the iX3. Detailed information on the relevant parameter configurations is described in the table below.

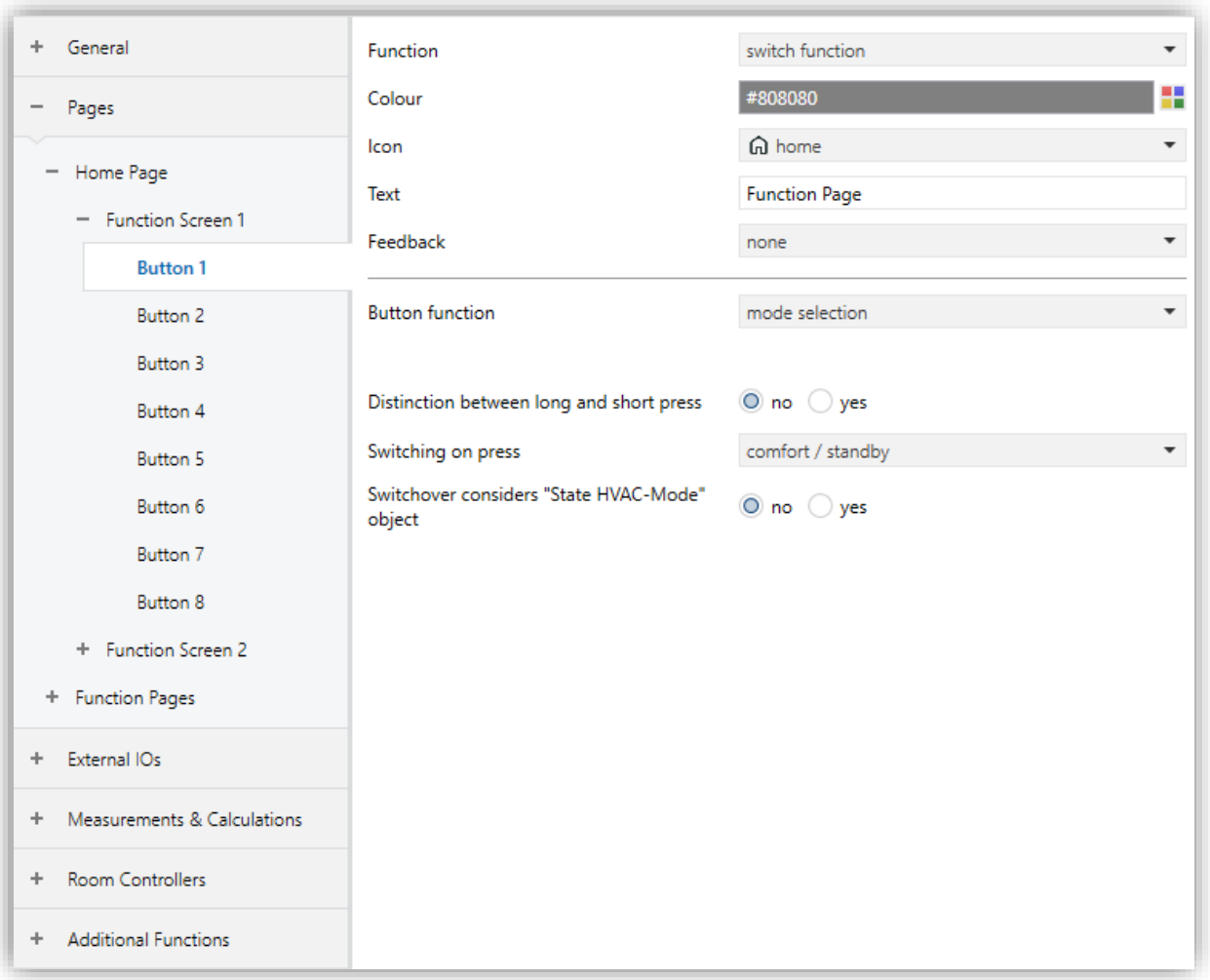


Fig. 17: Mode Selection Function Configuration

4.2.6.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Button Name	This parameter is used to type an input name. The name can be consisting of 40 characters	**40 Bytes allowed**
Button Function	This parameter is used to determine the Button X operation mode. If no function is selected, Button X will not be used. For other choices, all functionalities are configured separately.	No function Switch Switch/dimming Shutter/blinds Value/forced operation Scene control **Mode selection** Command sequence Counter RGB colour control RGBW control Thermostat Extension
Distinction between long and short press	This parameter is used to set if the input differentiates between short and long operations. With the option "yes", after opening/closing of the contract, it must, first of all, be ascertained if a short or long operation has occurred here. Only thereafter will a possible reaction be triggered.	**No** Yes
-> Switching on press[1]	A distinction is not made between short and long operations here. It is used to determine the press operation sending the value of the Button X.	**Comfort / standby** Comfort / economy Comfort / standby / economy Comfort / standby / economy / protection
-> Switching on short press[2]	This parameter is visible if there is a distinction between short and long operations. It is used to determine the short press operation sending the value of the Button X.	**Comfort / standby** Comfort / economy Comfort / standby / economy Comfort / standby / economy / protection
-> Reaction on long press[2]	This parameter is visible if there is a distinction between short and long operations. It is used to determine the long press operation sending the value of the Button X.	**Comfort** Standby Economy Protection

-> Long press after[2]	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	00:00.200… **00:00.500** … 01:05.535
Switchover considers "State HVAC-Mode" object	This parameter is used to enable the HVAC-Mode state object to change the current HVAC mode via KNX. If this parameter is selected as "Yes", the new value is sent according to feedback object's value. If feedback object doesn't update, the new value doesn't change.	**No** Yes

[1] This parameter is visible when the parameter "Distinction between long and short press" is set to "No".
[2] This parameter is visible when the parameter "Distinction between long and short press" is set to "Yes".

4.2.7. Command Sequence

In this section, it is explained how the command sequence function works. Up to 4 commands are attainable with either 1-bit, 1-byte (percentage) or 1-byte (0...255) objects. Each press event toggles through the used commands (Object A, B, C, D) via the assigned buttons. Detailed information on the relevant parameter configurations is described in the table below.

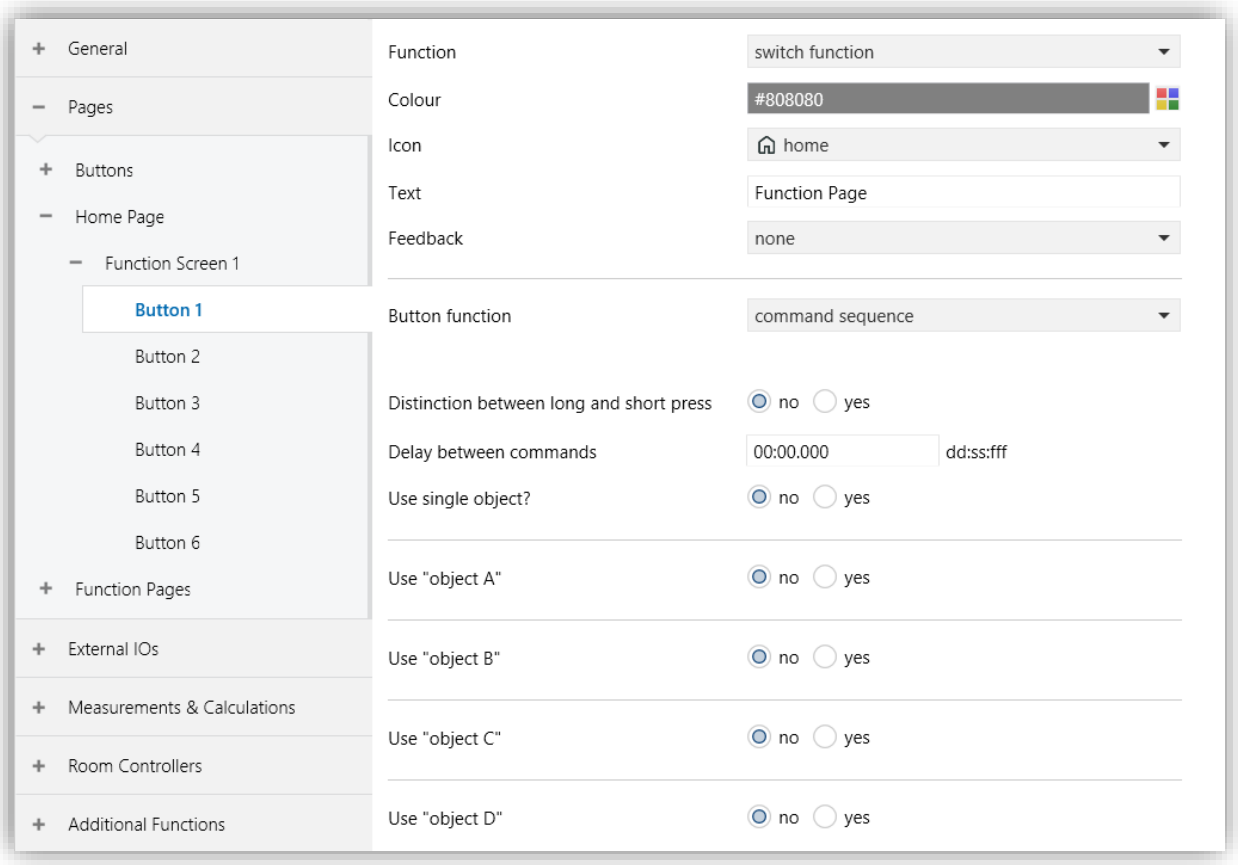


Fig. 18: Command Sequence Function Configuration

4.2.7.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Button name	This parameter is used to type an input name. The name can be consisting of 40 characters	**40 Bytes allowed**
Button function	This parameter is used to determine the Button X operation mode. If no function is selected, Button X will not be used. For other choices, all functionalities are configured separately.	No function Switch Switch/dimming Shutter/blinds Value/forced operation Scene control Mode selection **Command sequence** Counter RGB colour control RGBW control Thermostat Extension
Distinction between long and short press	This parameter is used to set if the input differentiates between short and long operations. With the option "yes", after opening/closing of the contract, it must, first of all, be ascertained if a short or long operation has occurred here. Only thereafter will a possible reaction be triggered.	**No** Yes
-> Long press after[1]	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	00:00.200… **00:00.500** … 01:05.535
Delay between commands	This parameter is used to determine the delay between sending the value of the sequence	**00:00.000**…00:20.000
Use single object?	This parameter decides whether each sequence is sent to a single object or multiple objects.	**No** Yes
-> Use "object X"[2]	This parameter is used to enable each command object when they are set to yes.	**No** Yes
-> Data type[2]	This parameter is used to determine the sending data type to the bus when an operation occurs.	1 bit 1 byte (0...255) 1 byte (0...100%) HVAC mode

-> Value 'X'[2]	This parameter is used to determine the sending value to the bus when a short operation occurs.	Values depend on DPT selection.
-> Value 'X' for long press[3]	This parameter is used to determine the sending value to the bus when a long operation occurs.	Values depend on DPT selection.
-> Value amount[4]	This parameter is used to determine the debounce time. Debouncing prevents unwanted multiple operations of the button, e.g., due to bouncing of the contact.	2 **3** 4
-> Data type[4]	This parameter is used to determine the sending value to the bus when a short operation occurs.	1 bit 1 byte (0...255) 1 byte (0...100%) HVAC mode
-> Value 'X'[4]	This parameter is used to determine the sending value to the bus when a short operation occurs.	Values depend on DPT selection.
-> Value 'X' for long press[5]	This parameter is used to determine the sending value to the bus when a long operation occurs.	Values depend on DPT selection.

[1] This parameter is visible when the parameter "Distinction between long and short press" is set to "Yes".
[2] This parameter is visible when the parameter "Use single object?" is set to "No".
[3] This parameter is visible when the parameter "Distinction between long and short press" is set to "Yes" and the parameter "Use single object?" is set to "No".
[4] This parameter is visible when the parameter "Use single object?" is set to "Yes".
[5] This parameter is visible when the parameters "Distinction between long and short press" and "Use single object?" are set to "Yes".

4.2.8. Counter

In this section, it is explained how to count input pulses on the iX3. Detailed information on the relevant parameter configurations is described in the table below.

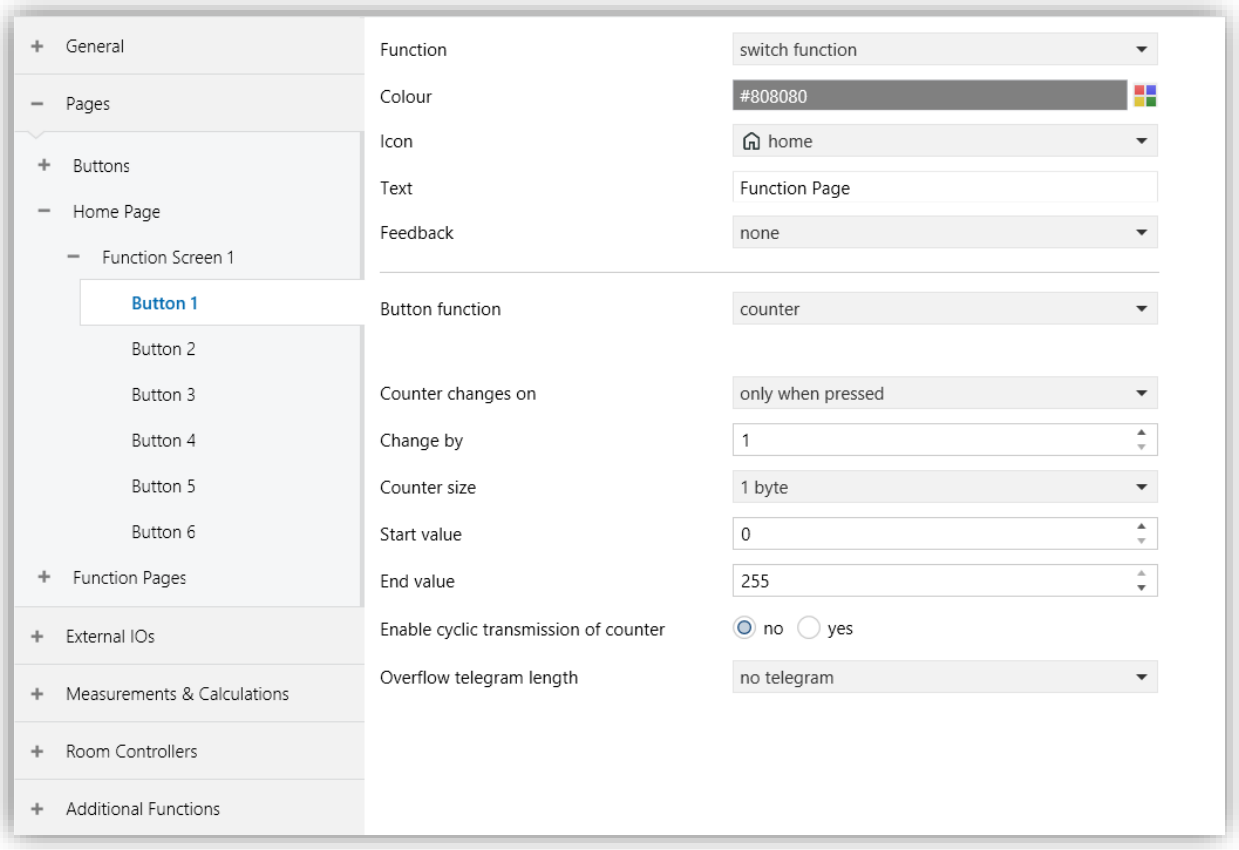


Fig. 19: Counter Function Configuration

4.2.8.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Button name	This parameter is used to type a button name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Button function	This parameter is used to determine the button function. If no function is selected, Button X will not be used. For other choices, all functionalities are configured separately.	No function Switch Switch/dimming Shutter/blinds Value/forced operation Scene control Mode selection Command sequence **Counter** RGB colour control RGBW control Thermostat Extension
Counter increase on	This parameter is used to set how the input pulse is to be generated.	**Only when pressed** Only when released Both when pressed and released
Change by	This parameter is used to assign the changing size when a press event occurs.	**1**..255
Counter size	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	**1 byte** 2 bytes 4 bytes
Start value	This parameter is used to set the initial value of the counter after a reset or failure.	Values depend on DPT selection.
End value	This parameter is used to set the end value of the counter.	Values depend on DPT selection.
Enable cyclic transmission of counter	This parameter is used to determine if the counter value is sent cyclically on the bus.	**No** Yes
-> Repeated transmit cycle period[1]	This parameter is used to determine the sending value to the bus when a short operation occurs.	00:00.200…**00:00.500** …01:05.535

-> Wait button trigger after reset[1]	This parameter is used to set startup behaviour of periodic sending of counter value. *Counter value starts from "Start value" parameter after reset.	**No** Yes
Overflow telegram length	This parameter is used to set the length of the overflow telegram which will be sent to the bus when the counter value exceeds the end value set in the parameter list.	**No telegram** 1 bit 1 byte
-> Overflow telegram value[2]	This parameter is used to determine the sending value to the bus when a short operation occurs.	Values depend on DPT selection.

[1] This parameter is visible when the parameter "Enable cyclic transmission of counter" is set to "Yes".
[2] This parameter is visible when the parameter "Overflow telegram length" is set to "1 bit" or "1 byte".

4.2.9. RGB Colour Control

This section, it is explained how to control an RGB LED device through the buttons connected to the iX3. Detailed information on the relevant parameter configurations is described in the table below.

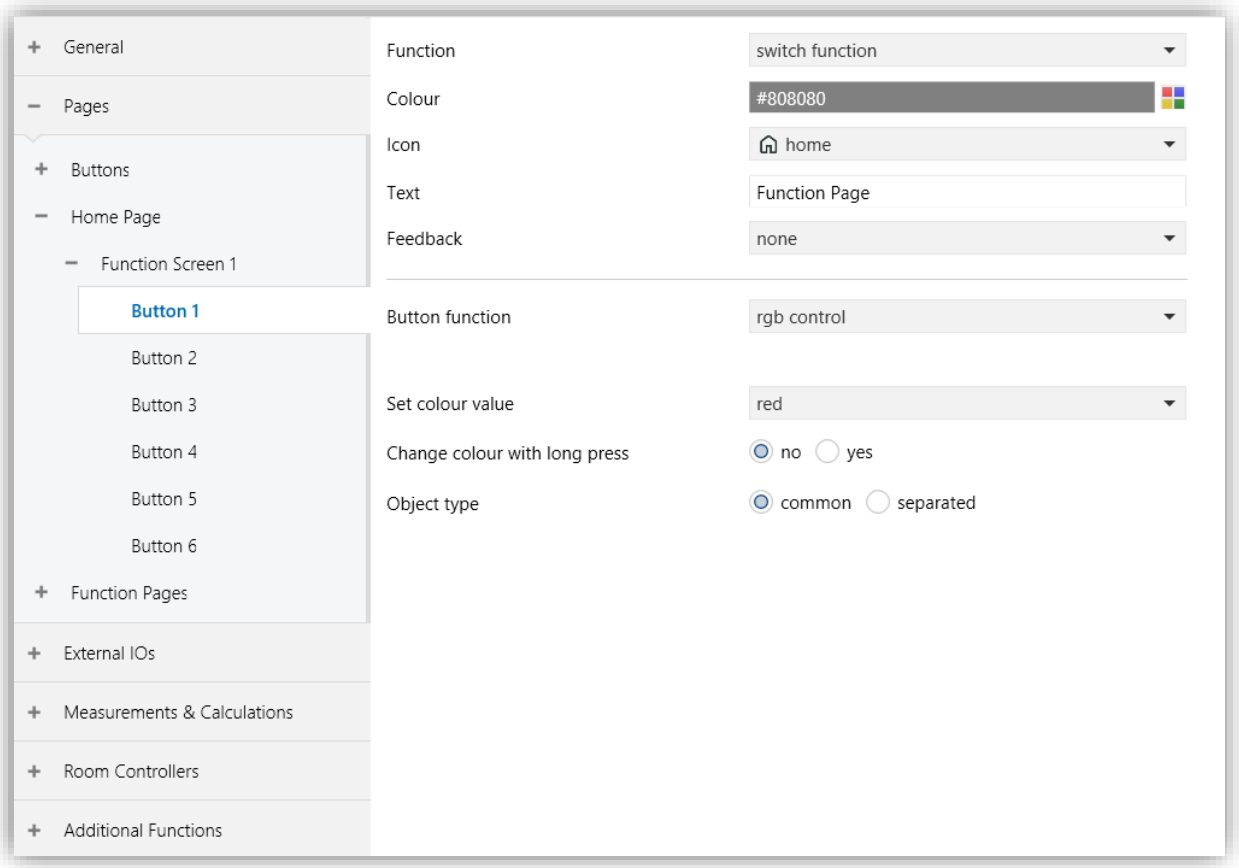


Fig. 20: RGB Colour Control Function Configuration

4.2.9.1. Parameters List

PARAMETER	DESCRIPTION	VALUES
Button name	This parameter is used to type a button name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Button Function	This parameter is used to determine the button function. If no function is selected, Button X will not be used. For other choices, all functionalities are configured separately.	No function Switch Switch/dimming Shutter/blinds Value/forced operation Scene control Mode selection Command sequence Counter **RGB colour control** RGBW control Thermostat Extension
Set colour value	This parameter is used to set RGB colours according to the configured values.	**Red** Orange Yellow Green-yellow Green Green-cyan Cyan Blue-cyan Blue Blue-magenta Magenta Red-magenta White
Change colour with long press	This parameter is used to enable or disable the colour changing with long press operation.	**No** Yes
-> Long press after[1]	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	00:00.200…**00:00.500** …01:05.535
Object type	This parameter is used to determine the RGB colour object value.	**common** separated

[1] This parameter is visible when the parameter "Change colour with long press" is set to "Yes".

4.2.10. RGBW Control

This section, it is explained how to control an RGBW device through the buttons connected to the iX3. Detailed information on the relevant parameter configurations is described in the table below.

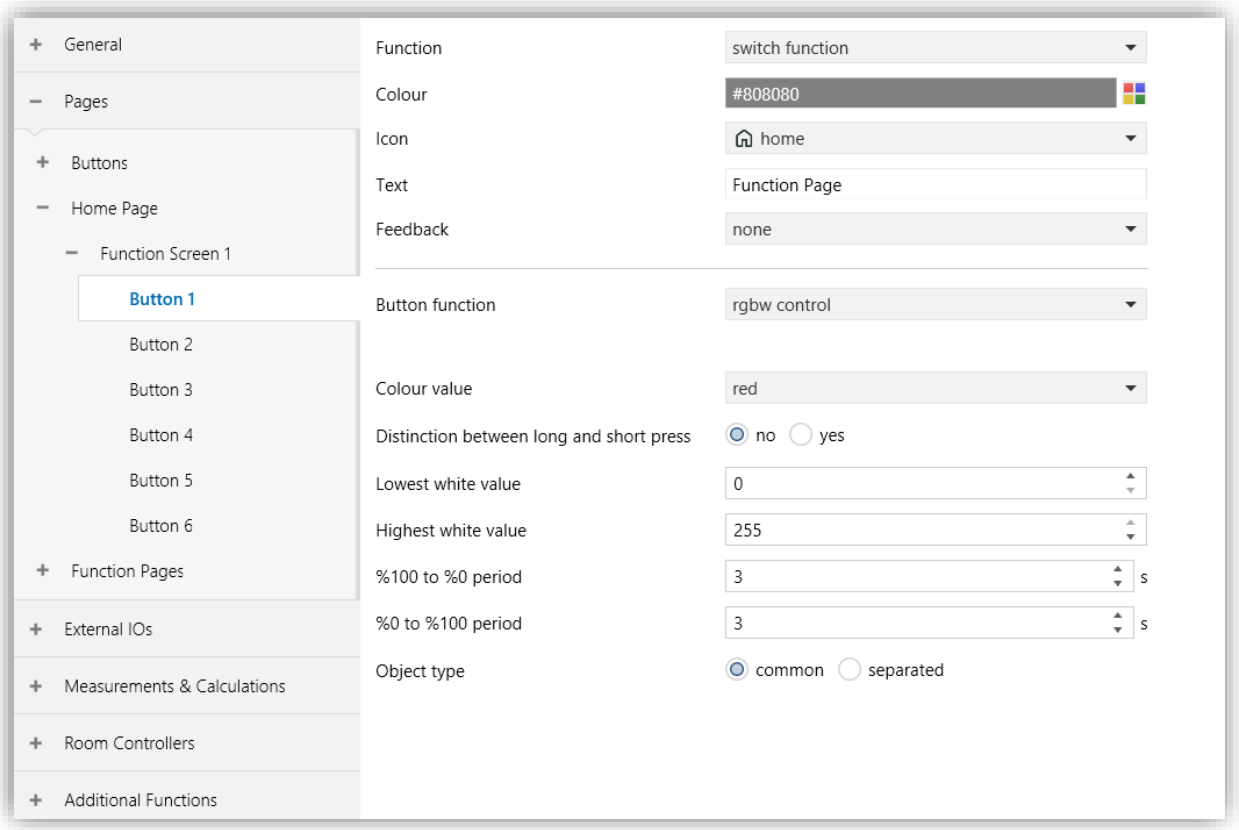


Fig. 21: RGBW Control Configuration Page

4.2.10.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Button name	This parameter is used to type a button name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Button function	This parameter is used to determine the button function. If no function is selected, Button X will not be used. For other choices, all functionalities are configured separately.	No function Switch Switch/dimming Shutter/blinds Value/forced operation Scene control Mode selection Command sequence Counter RGB colour control **RGBW control** Thermostat Extension
Colour value	This parameter is used to set RGBW colours according to the configured values.	**Red** Orange Yellow Green-yellow Green Green-cyan Cyan Blue-cyan Blue Blue-magenta Magenta Red-magenta White
Distinction between long and short press	This parameter is used to enable or disable the colour changing with long press operation.	**No** Yes

-> Long press after [1]	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	00:00.200…**00:00.500** …01:05.535
Lowest white value	This parameter is set to the lowest white value.	**0**…254
Highest white value	This parameter is set to the highest white value.	1…**255**
%100 to %0 period	This parameter is used to set how long it takes to go from 100% to 0%.	1s…**3s**…10s
%0 to %100 period	This parameter is used to set how long it takes to go from 0% to 100%.	1s…**3s**…10s
Object type	This parameter is used to determine the RGBW colour object type.	**common** separated

[1] This parameter is visible when the parameter "Distinction between long and short press" is set to "Yes".

4.2.11. Thermostat Extension

This section, it is explained how to control a thermostat device through the buttons connected to the iX3. Detailed information on the relevant parameter configurations is described in the table below.

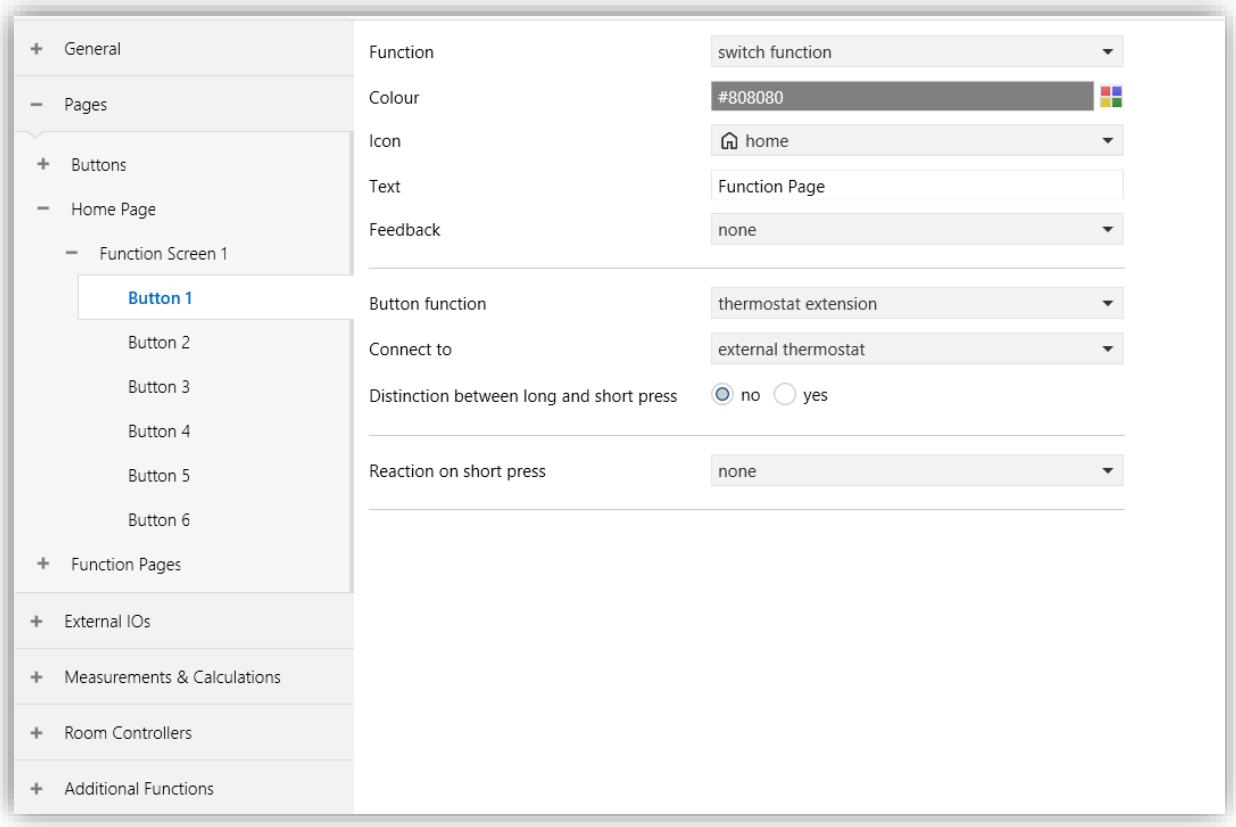


Fig. 22: Thermostat Extension Configuration Page

4.2.11.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Button name	This parameter is used to type a button name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Button function	This parameter is used to determine the button function. If no function is selected, Button X will not be used. For other choices, all functionalities are configured separately.	No function **Switch** Switch/dimming Shutter/blinds Value/forced operation Scene control Mode selection Command sequence Counter RGB colour control RGBW control **Thermostat Extension**
Connect to	This parameter selects whether the thermostat to be connected to the device is external or internal.	**External thermostat** Internal thermostat 1 Internal thermostat 2 Internal thermostat 3 Internal thermostat 4
Distinction between long and short press	This parameter is used to enable or disable the control changing with long press operation.	**No** Yes
-> **Reaction on long press**[1]	This parameter is used to determine the long press operation sending the value of the Button X.	**None** Status Control Heating cooling control HVAC mode control Setpoint control Fan control
-> **Long press after**[1]	This parameter is used to determine long operation detection after the button press operation. For	00:00.200…**00:00.500** …01:05.535

	making a long operation, the button should be pressed at least the configured value.	
Reaction on short press	This parameter is used to determine the short press operation sending the value of the Button X.	**None** Status Control Heating cooling control HVAC mode control Setpoint control Fan control
Reaction on short press / Reaction on long press: Status Control		
Status operation	This parameter is used to determine which status value will be sent for each long or short press operation. **Fixed:** Disable or Enable value will be sent according to the parameter that will be appear so the user can select the value. **Toggle:** On each short or long operation, toggled of the last status value will be sent.	**Fixed** Toggle
-> Status set value[2]	This parameter is used to determine the status value to be sent.	**Disable** Enable
-> Separate feedback object[3]	This parameter is used to activate the group object for status feedback.	**No** Yes
Reaction on short press / Reaction on long press: Heating cooling control		
Working mode operation	This parameter is used to determine which status value will be sent for each long or short press operation. **Fixed:** Cooling or Heating value will be sent according to a parameter that will be appear so the user can select the value. **Toggle:** On each short or long operation, toggled of the last working mode value will be sent.	**Fixed** Toggle
-> Working mode set value[4]	This parameter is used to determine the working mode value to be sent.	**Cooling** Heating
-> Separate feedback object[5]	This parameter is used to activate the group object for working mode feedback.	**No** Yes

Reaction on short press / Reaction on long press: HVAC mode control		
Mode operation	This parameter is used to determine which HVAC mode value will be sent for each long or short press operation. **Fixed:** HVAC mode value will be sent according to a parameter that will be appear so the user can select the value. **Toggle:** On each short or long operation, the next HVAC mode that was activated, will be sent.	**Fixed** Toggle
-> Mode set value[6]	This parameter is used to determine the HVAC mode value to be sent.	Auto **Comfort** Standby Economy Protection
-> Switch over modes[7]	This parameter is used to determine which HVAC modes will be sent sequentially.	**Comfort / standby** Comfort / economy Comfort / standby / economy Comfort / standby / economy/protection
-> Enable feedback object[7]	This parameter is used to activate the group object for HVAC mode feedback.	**No** Yes
Reaction on short press / Reaction on long press: Setpoint control		
Setpoint operation	This parameter is used to determine the setpoint value will be sent for each long or short press operation. **Fixed:** The setpoint value will be sent according to a parameter that will be appear so the user can select the value. **Decrease:** On each long or short operation the setpoint value will decrease step by step according to a parameter that will be appear so the user can select the step value. **Increase:** On each long or short operation the setpoint value will increase step by step according to	**Fixed** Decrease Increase

	a parameter that will be appear so the user can select the step value.	
-> Setpoint type[8]	This parameter is used to determine the setpoint data type.	**Individual** Dependent
-> Setpoint set value[8]	This parameter is used to determine the setpoint value to be sent.	**25.0°C** (10.0 … 40.0) **0.0°C** (-10.0 … 10.0)
-> Setpoint step[9]	This parameter is used to determine the step value for increasing or decreasing the setpoint value.	0.1K, 0.5K, 1K, 2K
-> Separate feedback object[9]	This parameter is used to activate the group object for setpoint value feedback.	**No** Yes
Reaction on short press / Reaction on long press: Fan control		
Fan control type	This parameter is used to determine which parameter of fan will be controlled.	**Fan level** Fan mode
-> Fan level operation[10]	This parameter is used to determine the fan level value will be sent for each long or short press operation. **Fixed:** The fan level will be sent according to a parameter that will be appear so the user can select the value. **Decrease:** On each long or short operation the fan level value will decrease step by step up to minimum level. **Increase:** On each long or short operation the fan level value will increase step by step up to maximum level. **Sequential:** On each long or short operation, the fan level value increases step by step up to the maximum level. After reaching the maximum level, it goes back to the minimum level again.	**Fixed** Decrease Increase Sequential
-> Fan level set value[11]	This parameter is used to determine the fan level value to be sent.	**0**…5
-> Fan max level[12]	This parameter is used to determine the maximum fan level of the external thermostat.	**0**…5
-> Fan mode control[13]	This parameter is used to determine which fan mode value will be sent for each long or short press operation.	**Fixed** Toggle

	Fixed: Fan mode value will be sent according to a parameter that will be appear so the user can select the value. **Toggle:** On each short or long operation, toggled of the last fan mode value will be sent.	
-> Fan mode set value[14]	This parameter is used to determine the fan mode value to be sent.	**Auto** Manual
-> Separate feedback object[12,15]	This parameter is used to activate the group object for fan level[12] and fan mode[15] value feedback.	**No** Yes

[1] This parameter is visible when the parameter "Distinction between long and short press" is set to "Yes".
[2] This parameter is visible when the parameter "Status operation" is set to "Fixed".
[3] This parameter is visible when the parameter "Status operation" is set to "Toggle" and connected to "External Thermostat".
[4] This parameter is visible when the parameter "Working mode operation" is set to "Fixed".
[5] This parameter is visible when the parameter "Working mode operation" is set to "Toggle" and connected to "External Thermostat".
[6] This parameter is visible when the parameter "Mode operation" is set to "Fixed".
[7] This parameter is visible when the parameter "Mode operation" is set to "Toggle" and connected to "External Thermostat".
[8] This parameter is visible when the parameter "Setpoint operation" is set to "Fixed".
[9] This parameter is visible when the parameter "Setpoint operation" is set to "Decrease" or "Increase".
[10] This parameter is visible when the parameter "Fan control type" is set to "Fan level".
[11] This parameter is visible when the parameter "Fan level operation" is set to "Fixed".
[12] This parameter is visible when the parameter "Fan level operation" is set to "Decrease" or "Increase" or "Sequential" and connected to "External Thermostat".
[13] This parameter is visible when the parameter "Fan control type" is set to "Fan mode".
[14] This parameter is visible when the parameter "Fan mode control" is set to "Fixed".
[15] This parameter is visible when the parameter "Fan mode control" is set to "Toggle" and connected to "External Thermostat".

4.2.12. Colour Control

This section, it is explained how to control colour through the colour control page in the iX3. Detailed information on the relevant parameter configurations is described in the table below.

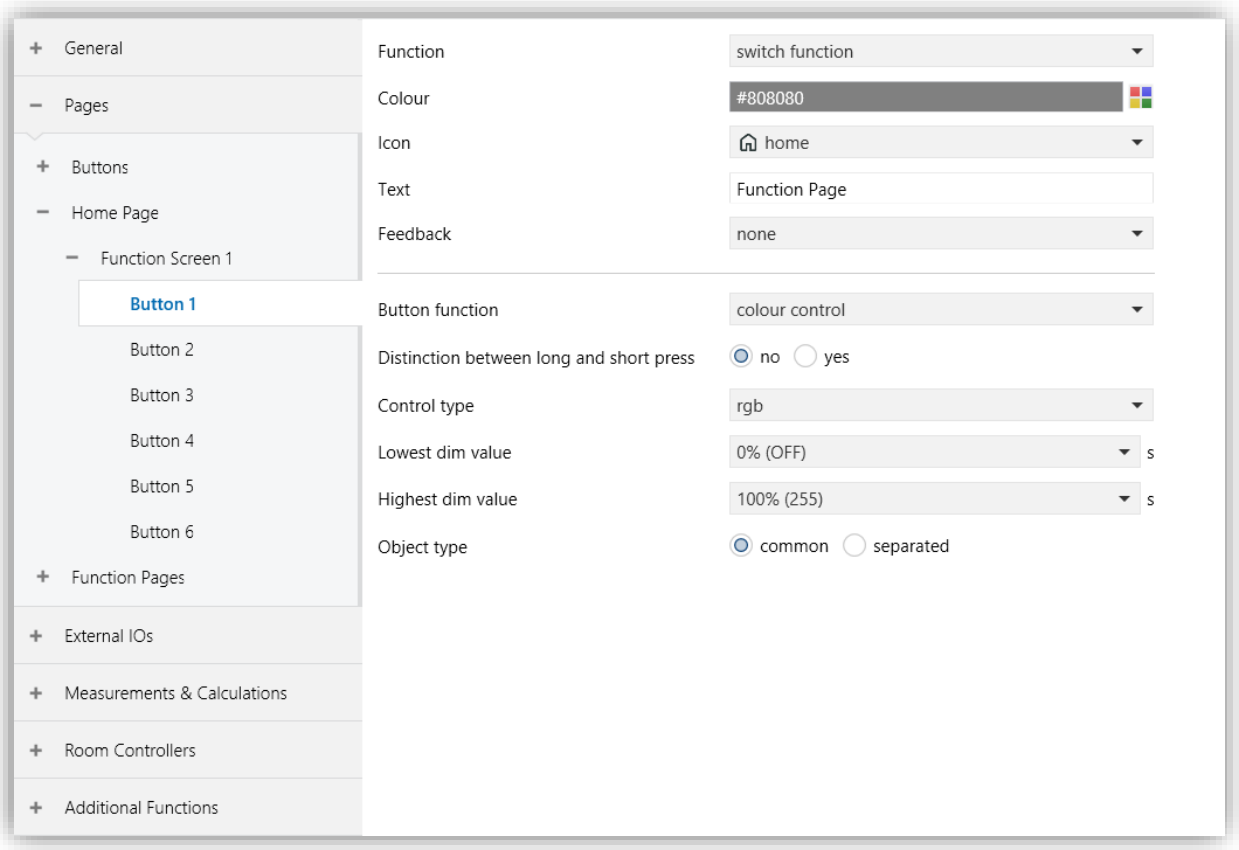


Fig. 23: Colour Control Configuration Page

4.2.12.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Distinction between long and short press	This parameter is used to enable or disable the colour changing with long press operation.	**No** Yes
-> Long press after[1]	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value. **NOTE:** If long press action is activated, after long press operation, colour control screen is shown. The switch function is triggered on short press. Long press action is not activated, colour control screen is shown on short press.	00:00.200…**00:00.500** …01:05.535
Control Type	This parameter determines the control type of colour control. According to this parameter colour control screen is configured.	RGB RGBW RGBW + Colour Temperature Brightness + Colour Temperature
Lowest dim value	This parameter determines the minimum dimming value.	**%0**…%100
Highest dim value	This parameter determines the maximum dimming value.	%1…**%100**
-> Lowest white value[2]	This parameter determines the minimum white value.	**%0**…%100
-> Highest white value[2]	This parameter determines the maximum white value.	%1…**%100**
-> Lowest colour temperature value[3]	This parameter determines the minimum colour temperature value.	**1000**…10000
-> Highest colour temperature value[3]	This parameter determines the maximum colour temperature value.	1000…**10000**
Object Type	This parameter is used to determine the colour control object value.	**Common** Separated

[1] This parameter is visible when the parameter "Distinction between long and short press" is set to "Yes".
[2] This parameter is visible when the parameter "Control type" is set to "RGBW" or "RGBW + colour temperature" or "Brightness + colour temperature"
[3] This parameter is visible when the parameter "Control type" is set to "RGBW + colour temperature" or "Brightness + colour temperature"

4.2.13. Music Control

This section, it is explained how to control music system through the music control page in the iX3. Music function is enabled, objects for background music control are visible, such as power on/off, play/pause, volume+/-, next song/previous song, play mode, music source, etc. Through these objects can control the music module. Detailed information on the relevant parameter configurations is described in the table below.

If the button function is selected as "Music Control", button's feedback only can be indicated "Power" and "Playing" status.

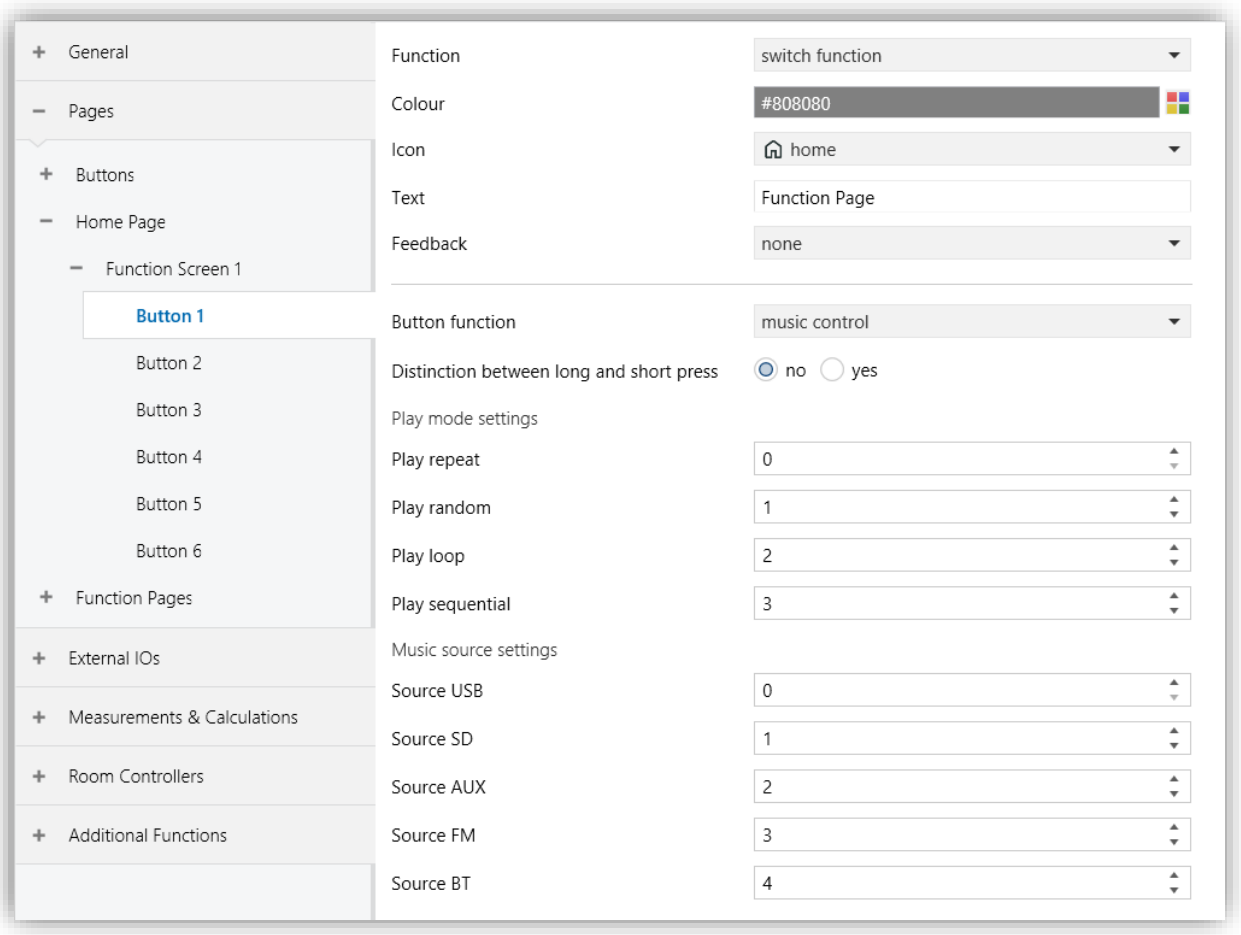


Fig. 24: Music Control Configuration Page

4.2.13.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Distinction between long and short press	This parameter is used to enable or disable the colour changing with long press operation.	**No** Yes
-> Short press action[1]	This parameter determines the which music event happens on short press action.	**Power off** Power on Power toggle Song play Song pause Song toggle Song previous Song next Volume up Volume down Mode repeat Mode random Mode loop Mode sequential Source USB Source SD Source AUX Source FM Source BT
-> Long press after[1]	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value. **NOTE:** If long press action is activated, after long press operation, music control screen is shown. Long press action is not activated, music control screen is shown on short press.	00:00.200…**00:00.500** …01:05.535

Play repeat	This parameter determines the output value for repeat play mode.	**0**…255
Play random	This parameter determines the output value for random play mode.	0…**1**…255
Play loop	This parameter determines the output value for loop play mode.	0…**2**…255
Play sequential	This parameter determines the output value for sequential play mode.	0…**3**…255
Source USB	This parameter determines the output value for USB music source.	**0**…255
Source SD	This parameter determines the output value for SD music source.	0…**1**…255
Source AUX	This parameter determines the output value for AUX music source.	0…**2**…255
Source FM	This parameter determines the output value for FM music source.	0…**3**…255
Source BT	This parameter determines the output value for BT music source.	0…**4**…255

[1] This parameter is visible when the parameter "Distinction between long and short press" is set to "Yes".

4.3. External Inputs/Outputs

This section, it is explained how to control the external inputs connected to the iX3. Digital or analog inputs can be connected to external inputs. If external input's type is selected as analog, it is considered a sensor. Therefore, the end-users can be configured the parameters below measurement channel. Temperature and brightness sensor can be connected to external inputs. Temperature and brightness measurements are made with these inputs.

If external input's type is selected as digital, the inputs are used as generic input with button functions such as switch, dimming, value forced etc. Additionally, window contact, presence input and card holder input can be used for energy-saving functions below the room controller channel.

External 1/2 inputs can be selected as digital or analog. External Input 3/4 can be selected only as digital.

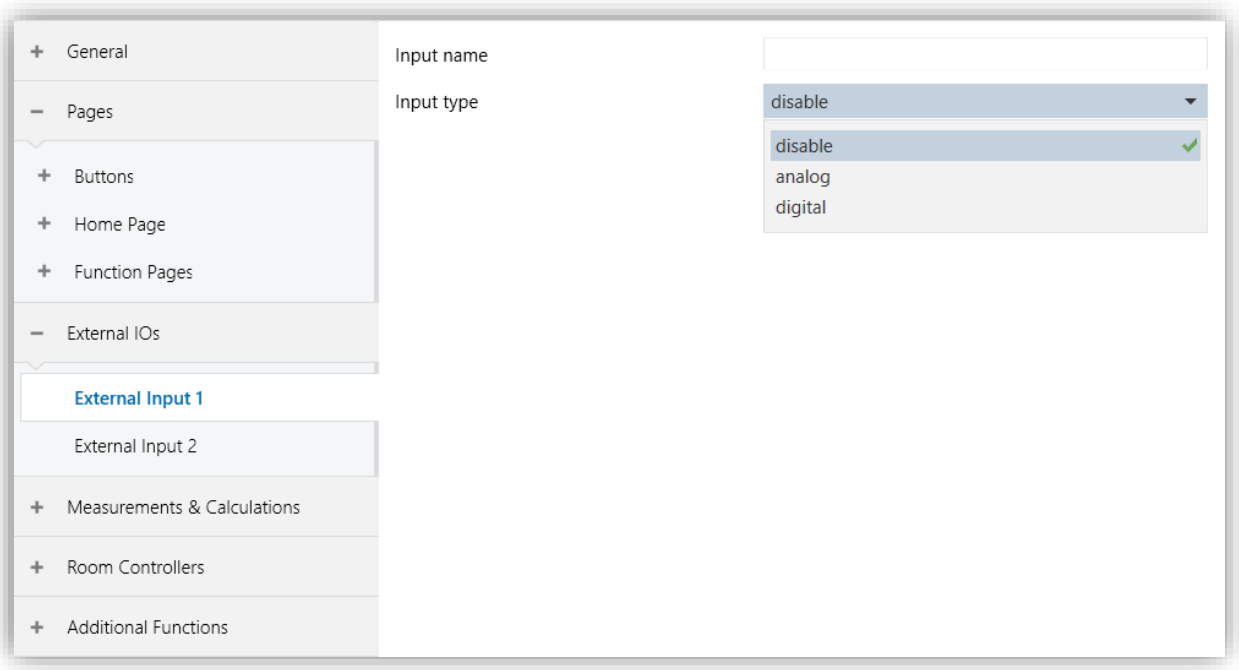


Fig. 25: External Inputs Page

4.3.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Input name	This parameter is used to type an Input name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Input type	This parameter is used to determine the button function. If disable is selected, the External Input X will not be used. For other choices, all functionalities are configured separately.	**Disable** Analog Digital
Input type	This parameter is used to determine the analog external input x functionality. In this section temperature functionality is described. **Temperature:** The input connected to the analog input is an NTC temperature sensor. **Brightness:** The input connected to the analog input is a light-dependent resistor (LDR) sensor.	Temperature Brightness

4.3.2. Analog Input – Temperature

This section describes how to configure a parameter for an NTC sensor that can be connected to the analog input of the iX3. After obtaining the necessary information about the NTC sensor to be connected from the relevant document, you should configure it.

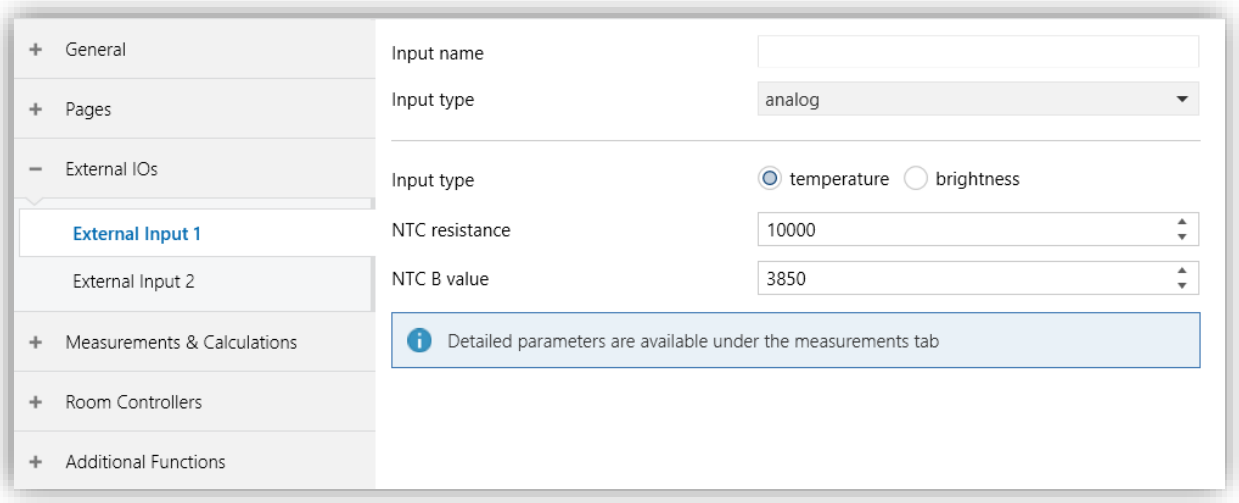


Fig. 26: Analog Input – Temperature Page

4.3.2.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Input name	This parameter is used to type an Input name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Input type	This parameter is used to determine the type of external input function. If disable is selected, the External Input X will not be used. For other choices, all functionalities are configured separately.	**Disable** Analog Digital
NTC resistance	This parameter is used to determine the resistance value of the NTC sensor to be used to measure the ambient temperature.	1…**10000**…65535
NTC B value	This parameter is used to determine the beta value of the NTC sensor to be used to measure the ambient temperature.	1…**3850**…65535

4.3.3. Analog Input – Brightness

This section describes how to configure a parameter for an LDR resistance that can be connected to the analog input of the iX3. After obtaining the necessary information about the LDR resistance to be connected from the relevant document, you should configure it.

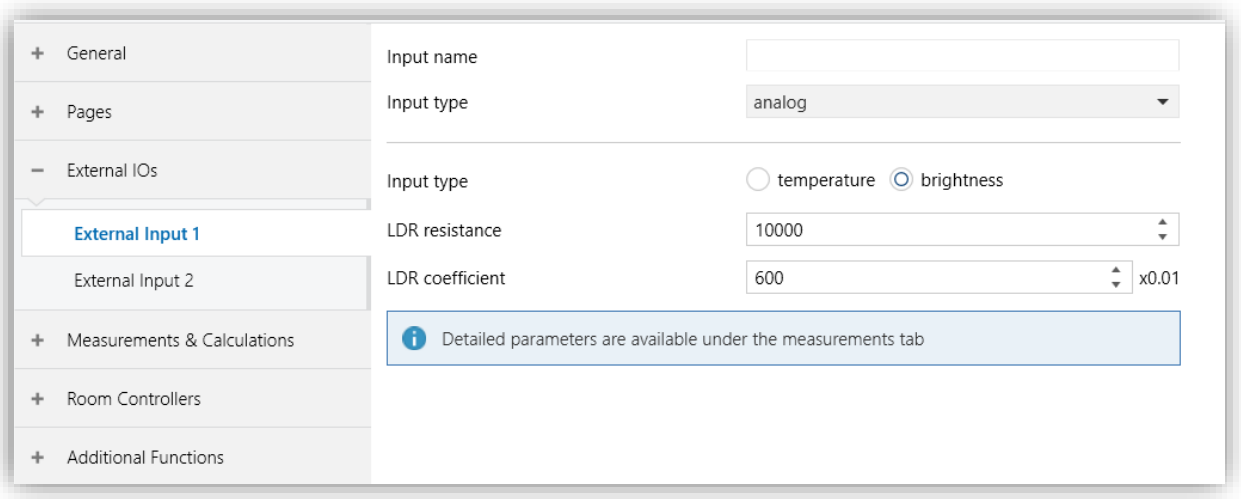


Fig. 27: Analog Input – Brightness Page

4.3.3.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Input name	This parameter is used to type an Input name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Input type	This parameter is used to determine the button function. If disable is selected, the External Input X will not be used. For other choices, all functionalities are configured separately.	**Disable** Analog Digital
Input type	This parameter is used to determine the analog external input x functionality. In this section temperature functionality is described. **Temperature:** The input connected to the analog input is an NTC temperature sensor. **Brightness:** The input connected to the analog input is a light-dependent resistor (LDR) sensor.	**Temperature** Brightness
LDR resistance	This parameter is used to determine the resistance value of the LDR to be used to measure the ambient brightness.	1…**10000**…65535
LDR coefficient (x 0.01)	This parameter is used to determine the coefficient value of the LDR to be used to measure the ambient brightness.	1…**600**…65535

4.3.4. Digital Input - Generic Input

This section describes how to configure a parameter for an external digital input that can be connected to the iX3. Detailed information on the relevant parameter configurations is described in the table below.

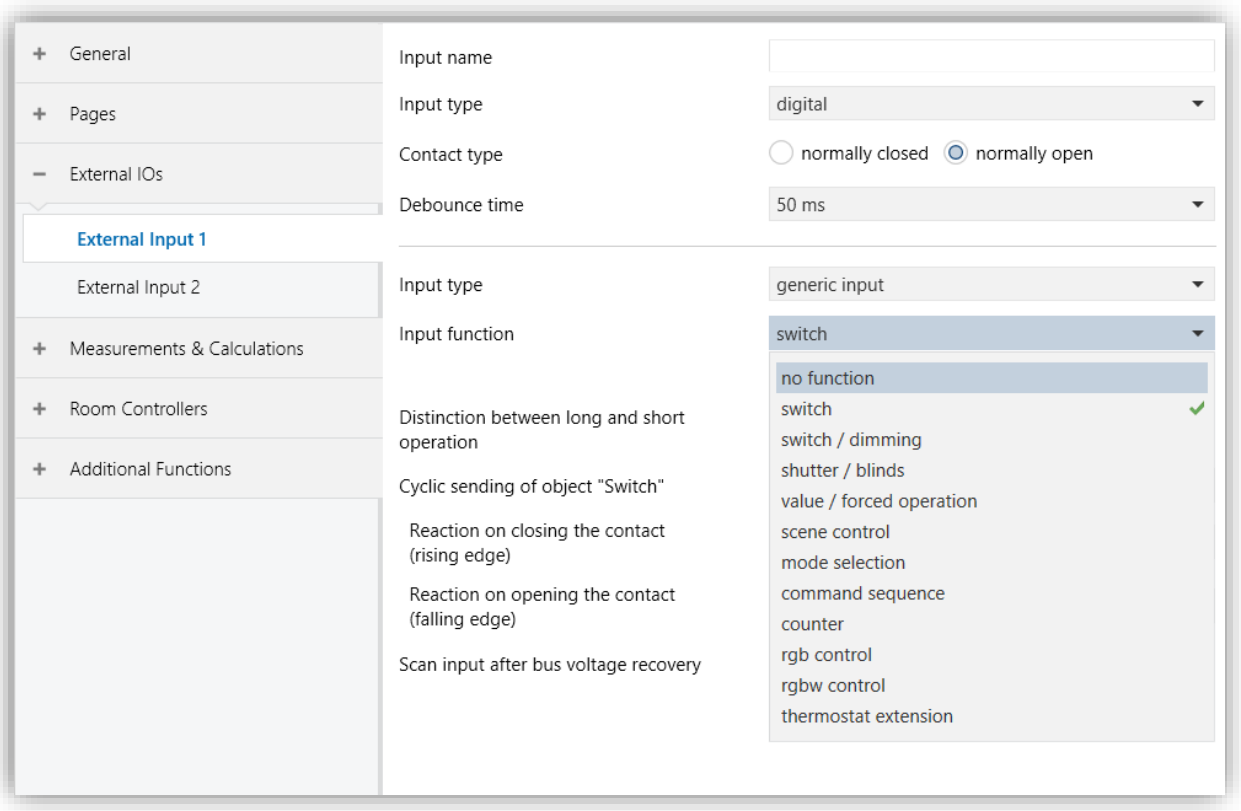


Fig. 28: Digital Input – Generic Input Page

4.3.4.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Input name	This parameter is used to type an Input name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Input type	This parameter is used to determine the type of external input function. If disable is selected, the External Input X will not be used. For other choices, all functionalities are configured separately.	**Disable** Analog Digital
Contact type	This parameter is used to specify the contact type that is connected to the iX3.	**Normally closed** Normally open
Debounce time	This parameter is used to determine the debounce time. Debouncing prevents unwanted multiple operations of the input, e.g., due to bouncing of the contact.	10ms 20ms 30ms 40ms **50ms** 70ms 100ms 150ms
Input type	This parameter is used to determine the input type. For other choices, all functionalities are configured separately.	**Generic input** Window contact Presence input Card holder
Input function	This parameter is used to determine the input function. If no function is selected, the input x will not be used. For other choices, all functionalities are configured separately.	**No function** Switch Switch/dimming Shutter/blinds Value/forced operation Scene control Mode selection Command sequence Counter RGB colour control RGBW control Thermostat Extension

4.3.5. Digital Input - Window Contact / Presence Input / Card Holder

This section describes how to configure a parameter for an external digital input such as window contact, presence input and card holder that can be connected to the iX3. Detailed information on the relevant parameter configurations is described in the table below.

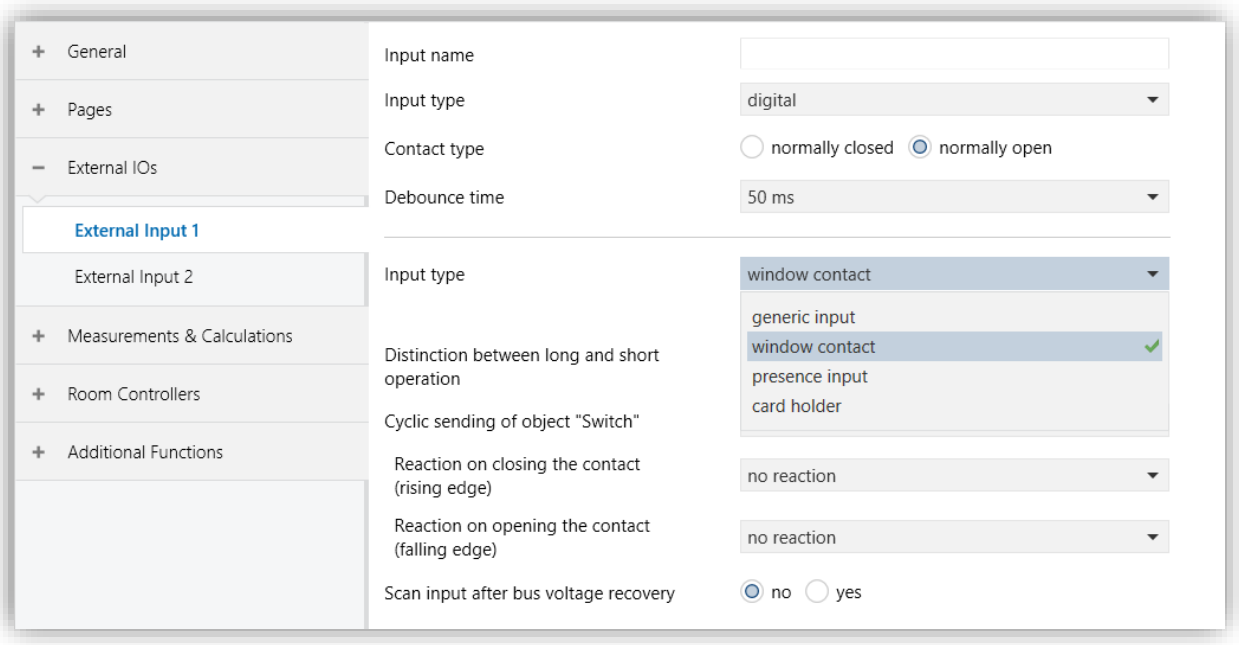


Fig. 29: Digital Input – Energy Saving Inputs

4.3.5.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Input name	This parameter is used to type an input name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Input type	This parameter is used to determine the type of external input function. If disable is selected, the External Input X will not be used. For other choices, all functionalities are configured separately.	**Disable** Analog Digital
Contact type	This parameter is used to specify the contact type that is connected to the iX3.	**Normally closed** Normally open
Debounce time	This parameter is used to determine the debounce time. Debouncing prevents unwanted multiple operations of the input, e.g., due to bouncing of the contact.	10 ms 20 ms 30 ms 40 ms **50 ms** 70 ms 100 ms 150 ms
Input type	This parameter is used to determine the input type. For other choices, all functionalities are configured separately.	**Generic input** Window contact Presence input Card holder
Distinction between long and short press	This parameter is used to set if the input differentiates between short and long operations. With the option "yes", after opening/closing of the contact, it must, first of all, be ascertained if a short or long operation has occurred here. Only thereafter will a possible reaction be triggered.	**No** Yes
Distinction between long and short press: No		
Cyclic sending of object "Switch"	This parameter is used to periodically send the commands to the bus line.	**No** If "Switch" = ON If "Switch" = OFF Always
-> Telegram repeated every[1]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams	00:00:01…**00:08:20**…18:12:15

Reaction on closing the contact (rising edge)	This parameter is visible if there is no distinction between short and long operations. For each edge, you can set if the object value is to be switched ON, OFF or TOGGLE, or if no reaction should occur. If cyclical sending has been parameterized, it is possible by setting the parameter value "terminate cyclic sending" with an operation of the input, to stop cyclic sending without a new object value being sent.	**No reaction** On Off Toggle
Reaction on opening the contact (falling edge)	This parameter is visible if there is no distinction between short and long operations. For each edge, you can set if the object value is to be switched ON, OFF or TOGGLE, or if no reaction should occur. If cyclical sending has been parameterized, it is possible by setting the parameter value "terminate cyclic sending" with an operation of the input, to stop cyclic sending without a new object value being sent.	**No reaction** On Off Toggle
Send button value after bus voltage recovery	This parameter is used to determine the sending value of the inputs when the bus voltage has been recovered.	**No** Yes
Distinction between long and short press: Yes		
Reaction on short press	This parameter is used to determine the short press operation sending the value of the input x.	**No reaction** On Off Toggle
Reaction on long press	This parameter is used to determine the long press operation sending the value of the input x.	**No reaction** On Off Toggle
Long press after	This parameter is used to determine long operation detection after the button press operation. For making a long operation, the button should be pressed at least the configured value.	00:00.200…**00:00.500** …01:05.535
Number of object for short/long press	This parameter is used to determine the object count to use for short and long operations. **1 object:** short and long operations will proceed with the same object. **2 objects:** short and long operations will proceed with 2 different objects.	**1 object** 2 objects

[1] This parameter is visible when the parameter "Cyclic sending of object "Switch"" is set to "If "Switch" = ON" or "If "Switch" = OFF" or "Always".

4.5. Measurement

The measurement channel folder includes the following sensors.

- Temperature Internal
- Humidity Internal
- Air Quality Internal
- Brightness Internal
- External 1 (Brightness / Temperature)
- External 2 (Brightness / Temperature)

The end-users can be configured the parameters related to the sensors given above. The sensor values can periodically be sent on the bus with a specified transmission interval, and whenever a specified variation occurs. Each sensor can be calibrated via a parameter or group object.

Thanks to the "Sampling rate" parameter, the end-users can be configured the updating interval of the channel value and additionally, the value filters such as median or low pass, are applied to the channel value for measurement noises. For example; if the filter type is median and the sampling rate is 10 seconds. The filtered value is updated per 10 seconds.

The median filter calculates an average with a series of measured values before sending on the bus. The parameter can have the following values:

- low = average value every 5 measurements;
- medium = average value every 15 measurements;
- high = average value every 25 measurements.

The low pass filter calculates and average with new measured values and previous measured value according to the following values:

- low = output value relies on new measurement more.
- medium = output value relies on new and previous measurements equally.
- high = output value relies on the previous measurements more.

Each sensor has an "Additional function". This feature provides to send the additional value to the KNX bus according to configured threshold levels.

Each sensor checks the bus healthy internally. If any error occurs, an alarm object is sent to the KNX bus to indicate that an error has occurred. Additionally, the error icons of the sensor are displayed on LCD screen. The error codes are going to explain in "Display Page".

4.5.1. Temperature Internal

This section describes how to configure the parameters for the internal temperature sensor of the iX3. The integrated temperature sensor allows the measuring of the room temperature in the range from -40 °C to +125 °C with a resolution of 0.2 °C.

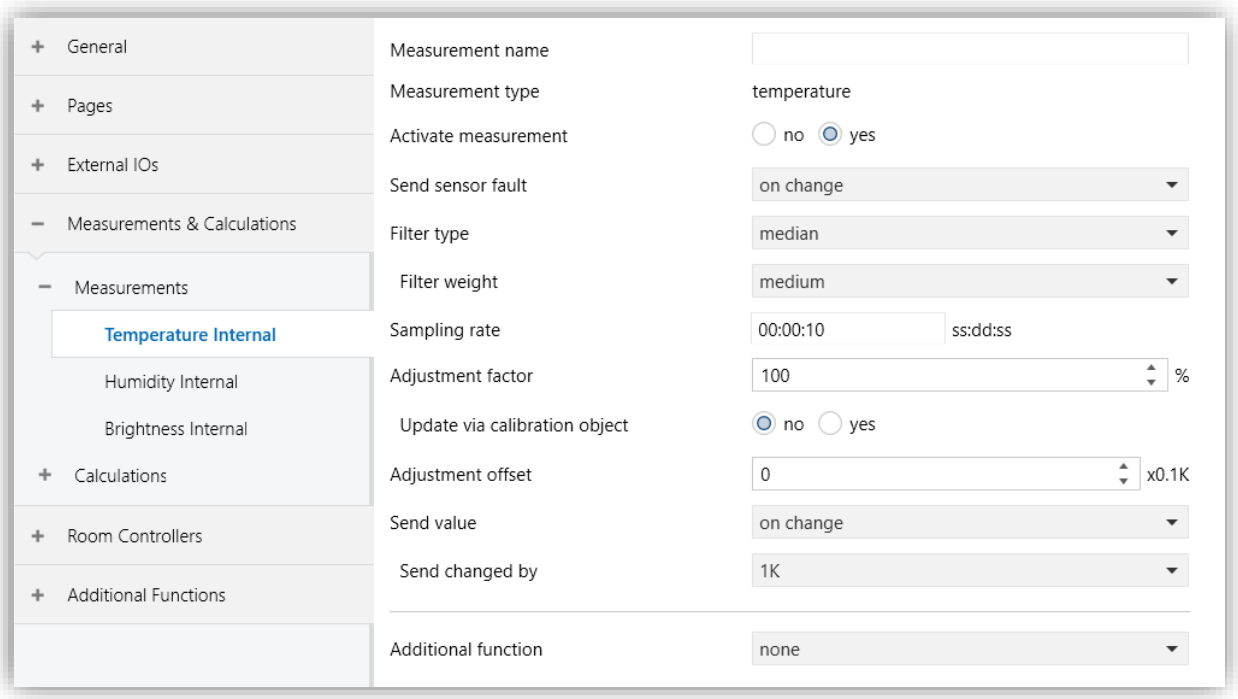


Fig. 31: Temperature Internal Page

4.5.1.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Measurement name	This parameter is used to type a Measurement name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Activate measurement	This parameter is used to enable or disable the measurement.	**No** Yes
Activate measurement: Yes		
Send sensor fault	This parameter allows sending the sensor fault information. **On change:** The sensor fault information is only sent when it changed. **Cyclic:** The sensor fault information is sent periodically. **On change and cyclic:** The value is sent both on change and cyclic.	Disable **On change** Cyclic On change & cyclic
-> Send cycle time[1]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams.	00:00:01 … **00:10:00** … 18:12:15
Filter type	This parameter is determined the type of sensor noise filter. **Median**: This filter calculates an average with a series of measured values before sending on the bus. **Low pass**: This filter calculates a value via 1st order IIR filter before sending on the bus.	None **Median** Low pass
-> Filter weight[2]	The parameter is determined the coefficient of the filter. **If median filter is selected;** **Low:** Average value every **5** measurements; **Medium:** Average value every **15** measurements; **High:** Average value every **25** measurements. **If low pass filter is selected;** **Low:** Output value relies on new measurement; **Medium:** output value relies on new and previous measurements equally.	Low **Medium** High

	High: output value relies on the previous measurements more	
Sampling rate	The parameter is determined the sampling time of the sensor. E.g., sampling rate is selected as 00:00:10, the sensor value is updated per 10 seconds.	00:00:01 … **00:00:10** … 18:12:15
Adjustment factor (%)	This parameter determines the calibration factor. This parameter can be changed on runtime via group object. In this case, the value measured by the sensor is multiplied by 0.01 of the set adjustment factor. Adjustment factor value can be calculated by this formula: Adjustment factor = (The real value that is read from external sensor / device value that is measured internally) × 100	0…**100**...65535
Update via calibration object	If this parameter is set to "**Yes"**, sensor calibration is carried out either via an object.	**No** Yes
Adjustment offset (x0.1K)	This parameter is used to determine the calibration value of the sensor.	-200... **0** …200
Send value	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission. **Cyclic:** "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	Disable **On change** Cyclic On change & cyclic
-> Send changed by[3]	This parameter determines the minimum variation for the sensor value to send the object.	0.1K, 0.2K, 0.3K, 0.5K, **1K**, 1.5K, 2K, 2.5K, 3K, 3.5K, 4K, 4.5K, 5K, 7.5K, 10K
-> Send cycle time[4]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams	00:00:01 … **00:00:10** … 18:12:15

Additional function	This parameter is used to determine the additional function of sensor measurement besides sending its value. If "Alarm function" is selected, low-level alarm and high-level alarm can be transmitted to bus via an object. Otherwise, a specific value can be transmitted via object with specific type.	**None** Send alarm Send bit Send byte Send Scene Send Percentage
Low level threshold (x0.1K)[5]	This parameter determines the low-level value of the additional function. The low threshold must be less than the high threshold.	-300…**0**…700
High level threshold (x0.1K)[5]	This parameter determines the high-level value of the additional function. The high threshold must be higher than the low threshold.	-300…**0**…700
Threshold hysteresis (x0.1K)[5]	This parameter determines the hysteresis value of the additional function.	-200…**0**…200
Send low level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send low level value[7]	The value to be sent when the measurement value is lower than low-level threshold.	Values depend on DPT selection.
Send normal level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send normal level value[8]	The value to be sent when the measurement value is between low-level and high-level threshold.	Values depend on DPT selection.
Send high level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send high level value[9]	The value to be sent when the measurement value is higher than low-level threshold.	Values depend on DPT selection.
Send alarm[5]	This parameter determines whether and when the value will be sent via an object.	Disable On change

	On change: "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission. **Cyclic:** "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	Cyclic On change & cyclic
-> Send cycle time[4]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams	00:00:01 … **00:00:10** … 18:12:15

[1] This parameter is visible when the parameter "Send sensor fault" is set to "Cyclic" or "On change & cyclic" or "Always".
[2] This parameter is visible when the parameter "Filter type" is set to "Median" or "Low pass".
[3] This parameter is visible when the parameter "Send value" is set to "On change" or "On change & cyclic".
[4] This parameter is visible when the parameter "Send value" is set to "Cyclic" or "On change & cyclic".
[5] This parameter is visible when the parameter "Additional function" is set to "Send alarm" or "Send bit" or "Send byte" or "Send scene" or "Send percentage". If the low threshold value is higher than the high threshold value and current value is exceed (high threshold) or dropped below (low threshold) the values, just low-level alarm value is sent over "Additional Value" object.
[6] This parameter is visible when the parameter "Additional function" is set to "Send bit" or "Send byte" or "Send scene" or "Send percentage".
[7] This parameter is visible when the parameter "Send bit > Send low-level alarm" is set to "Yes".
[8] This parameter is visible when the parameter "Send bit > Send normal-level alarm" is set to "Yes".
[9] This parameter is visible when the parameter "Send bit > Send high-level alarm" is set to "Yes".

4.5.2. Humidity Internal

This section describes how to configure the parameters for the internal humidity sensor of the iX3. The integrated relative humidity sensor allows the measuring of the relative humidity value in the room in the range from 0 %RH to 100 %RH with a resolution of 1.8 %RH. The measured value allows you to make an advanced room thermoregulation and enlarge the opportunities for a safe operation of certain types of terminal equipment used for cooling.

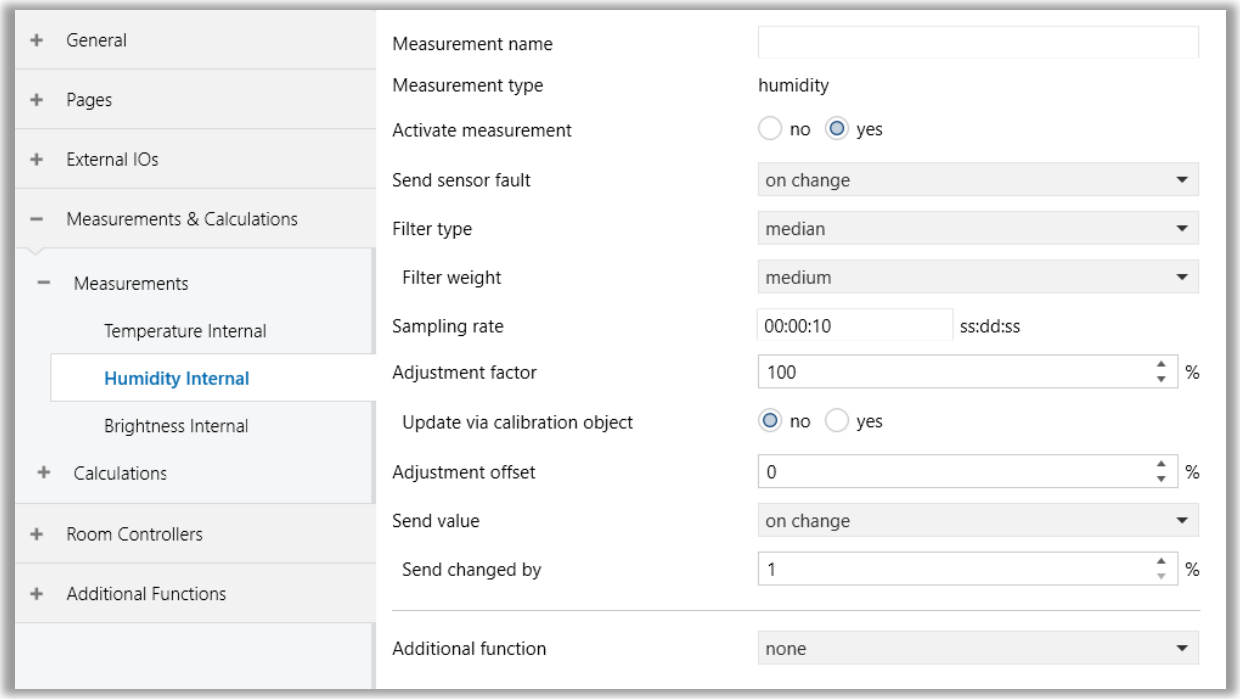


Fig. 32: Humidity Internal Page

4.5.2.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Measurement name	This parameter is used to type a Measurement name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Activate measurement	This parameter is used to enable or disable the measurement.	**No** Yes
Activate measurement: Yes		
Send sensor fault	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission. **Cyclic:** "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	Disable **On change** Cyclic On change & cyclic
-> Send cycle time[1]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams.	00:00:01 … **00:10:00** … 18:12:15
Filter type	This parameter is determined the type of sensor noise filter. **Median**: This filter calculates an average with a series of measured values before sending on the bus. **Low pass**: This filter calculates a value via 1st order IIR filter before sending on the bus.	None **Median** Low pass
-> Filter weight[2]	The parameter is determined the coefficient of the filter. **If median filter is selected;** **Low** = average value every **5** measurements; **Medium** = average value every **15** measurements; **High** = average value every **25** measurements. **If low pass filter is selected;** **Low** = output value relies on new measurement; **Medium** = output value relies on new and previous measurements equally.	Low **Medium** High

	High = output value relies on the previous measurements more	
Sampling rate	The parameter is determined the sampling time of the sensor. For example, sampling rate is selected as 00:00:10, the sensor value is updated per 10 seconds.	00:00:01 … **00:00:10** … 18:12:15
Adjustment factor (%)	This parameter determines the calibration factor. This parameter can be changed on runtime via group object. In this case, the value measured by the sensor is multiplied by 0.01 of the set adjustment factor. Adjustment factor value can be calculated by this formula: Adjustment factor = (The real value that is read from external sensor / device value that is measured internally) × 100	0…**100**...65535
Update via calibration object	If this parameter is set to "**Yes"**, sensor calibration is carried out either via an object.	**No** Yes
Adjustment offset (%)	This parameter is used to determine the calibration value of the sensor.	-40…**0**…40
Send value	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission. **Cyclic:** "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	Disable **On change** Cyclic On change & cyclic
Send changed by (%)[3]	This parameter determines the minimum variation for the sensor value to send the object.	0…**1**…40
-> Send cycle time[4]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams.	00:00:01 … **00:00:10** … 18:12:15
Additional function	This parameter is used to determine the additional function of sensor measurement besides sending its value.	**None** Send alarm Send bit

	If "Alarm function" is selected, low-level alarm and high-level alarm can be transmitted to bus via an object. Otherwise, a specific value can be transmitted via object with specific type.	Send byte Send Scene Send Percentage
Low level threshold (%)[5]	This parameter determines the low-level value of the additional function. The low threshold must be less than the high threshold.	0…**30**…100
High level threshold (%)[5]	This parameter determines the high-level value of the additional function. The high threshold must be higher than the low threshold.	0…**60**…100
Threshold hysteresis (%)[5]	This parameter determines the hysteresis value of the additional function.	0…**1**…100
Send low level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send low level value[7]	The value to be sent when the measurement value is lower than low-level threshold.	Values depend on DPT selection.
Send normal level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send normal level value[8]	The value to be sent when the measurement value is between low-level and high-level threshold.	Values depend on DPT selection.
Send high level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send high level value[9]	The value to be sent when the measurement value is higher than low-level threshold.	Values depend on DPT selection.
Send alarm[5]	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission.	Disable On change Cyclic On change & cyclic

	Cyclic: "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	
-> Send cycle time[4]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams	00:00:01 … **00:00:10** … 18:12:15

[1] This parameter is visible when the parameter "Send sensor fault" is set to "Cyclic" or "On change & cyclic" or "Always".
[2] This parameter is visible when the parameter "Filter type" is set to "Median" or "Low pass".
[3] This parameter is visible when the parameter "Send value" is set to "On change" or "On change & cyclic".
[4] This parameter is visible when the parameter "Send value" is set to "Cyclic" or "On change & cyclic".
[5] This parameter is visible when the parameter "Additional function" is set to "Send alarm" or "Send bit" or "Send byte" or "Send scene" or "Send percentage". If the low threshold value is higher than the high threshold value and current value is exceed (high threshold) or dropped below (low threshold) the values, just low-level alarm value is sent over "Additional Value" object.
[6] This parameter is visible when the parameter "Additional function" is set to "Send bit" or "Send byte" or "Send scene" or "Send percentage".
[7] This parameter is visible when the parameter "Send bit > Send low-level alarm" is set to "Yes".
[8] This parameter is visible when the parameter "Send bit > Send normal-level alarm" is set to "Yes".
[9] This parameter is visible when the parameter "Send bit > Send high-level alarm" is set to "Yes".

4.5.3. Air Quality Internal

This section describes how to configure the parameters for the internal air quality sensor of the iX3.

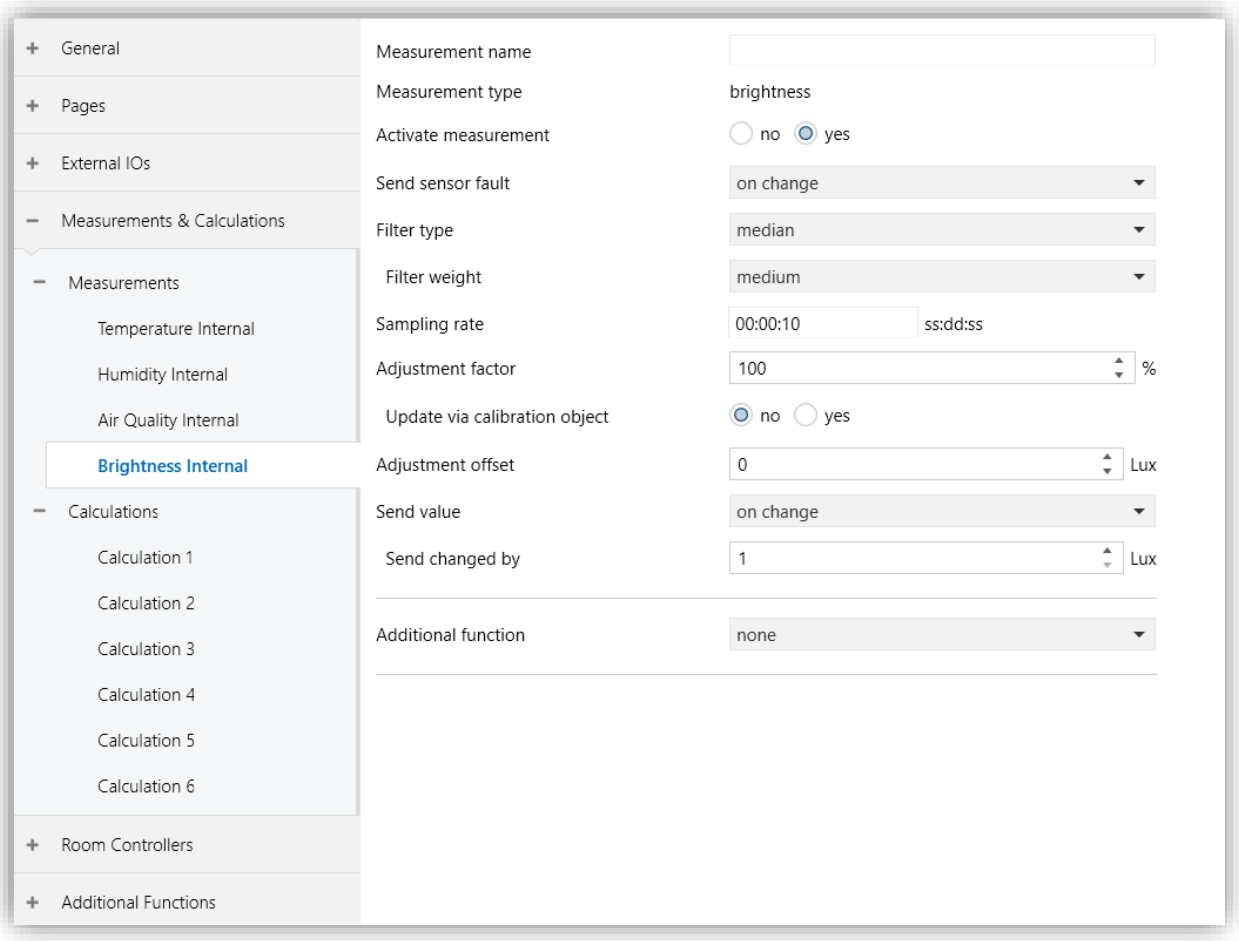


Fig. 33: Air Quality Internal Page

The integrated air quality sensor allows the measuring of the air quality value in the room by unit of VOC index. VOC Index is referenced to the average of VOCs present over the last 24 h in the room and notifies end users or air treatment devices when air pollution changes. Notifications are actionable in environments with low and high VOC backgrounds independent of the absolute VOC concentrations.

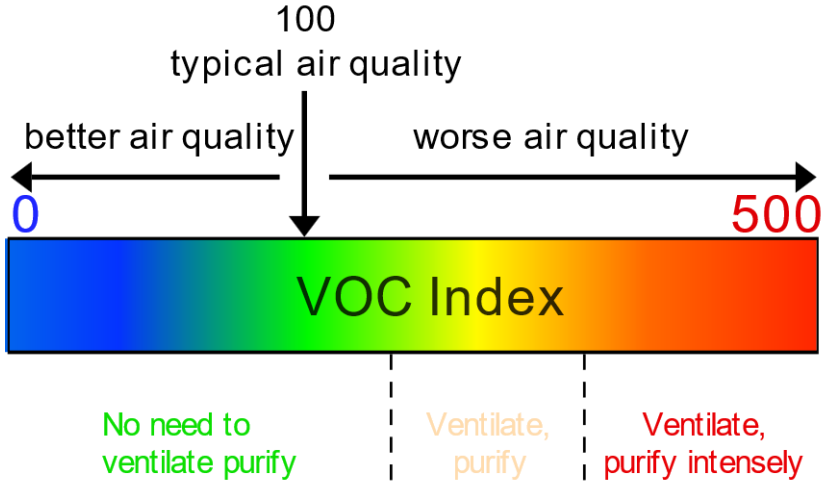


Fig. 34: Interpretation of Scaling

4.5.3.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Measurement name	This parameter is used to type a Measurement name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Activate measurement	This parameter is used to enable or disable the measurement.	**No** Yes
Activate measurement: Yes		
Send sensor fault	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission. **Cyclic:** "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	Disable **On change** Cyclic On change & cyclic
-> Send cycle time[1]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams.	00:00:01 … **00:10:00** … 18:12:15
Filter type	This parameter is determined the type of sensor noise filter. **Median**: This filter calculates an average with a series of measured values before sending on the bus. **Low pass**: This filter calculates a value via 1st order IIR filter before sending on the bus.	None **Median** Low pass
-> Filter weight[2]	The parameter is determined the coefficient of the filter. **If median filter is selected;** **Low** = average value every **5** measurements; **Medium** = average value every **15** measurements; **High** = average value every **25** measurements. **If low pass filter is selected;** **Low** = output value relies on new measurement; **Medium** = output value relies on new and previous measurements equally.	Low **Medium** High

	High = output value relies on the previous measurements more	
Sampling rate	The parameter is determined the sampling time of the sensor. For example, sampling rate is selected as 00:00:10, the sensor value is updated per 10 seconds.	00:00:01 … **00:00:10** … 18:12:15
Adjustment factor (%)	This parameter determines the calibration factor. This parameter can be changed on runtime via group object. In this case, the value measured by the sensor is multiplied by 0.01 of the set adjustment factor. Adjustment factor value can be calculated by this formula: Adjustment factor = (The real value that is read from external sensor / device value that is measured internally) × 100	0…**100**...65535
Update via calibration object	If this parameter is set to "**Yes"**, sensor calibration is carried out either via an object.	**No** Yes
Adjustment offset (ppm)	This parameter is used to determine the calibration value of the sensor.	-32768…**0**…32767
Send value	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission. **Cyclic:** "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	Disable **On change** Cyclic On change & cyclic
-> Send changed by (ppm)[3]	This parameter determines the minimum variation for the sensor value to send the object.	**1**…255
-> Send cycle time[4]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams.	00:00:01 … **00:00:10** … 18:12:15
Additional function	This parameter is used to determine the additional function of sensor measurement besides sending its value.	**None** Send alarm Send bit

	If "Alarm function" is selected, low-level alarm and high-level alarm can be transmitted to bus via an object. Otherwise, a specific value can be transmitted via object with specific type.	Send byte Send Scene Send Percentage
Low level threshold (ppm)[5]	This parameter determines the low-level value of the additional function. The low threshold must be less than the high threshold.	0…**100**…1200
High level threshold (ppm)[5]	This parameter determines the high-level value of the additional function. The high threshold must be higher than the low threshold.	0…**300**…1200
Threshold hysteresis (ppm)[5]	This parameter determines the hysteresis value of the additional function.	0…**80**…1200
Send low level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send low level value[7]	The value to be sent when the measurement value is lower than low-level threshold.	Values depend on DPT selection.
Send normal level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send normal level value[8]	The value to be sent when the measurement value is between low-level and high-level threshold.	Values depend on DPT selection.
Send high level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send high level value[9]	The value to be sent when the measurement value is higher than low-level threshold.	Values depend on DPT selection.
Send alarm[5]	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission.	Disable **On change** Cyclic On change & cyclic

	Cyclic: "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	
-> Send cycle time[4]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams	00:00:01 … **00:00:10** … 18:12:15

[1] This parameter is visible when the parameter "Send sensor fault" is set to "Cyclic" or "On change & cyclic" or "Always".
[2] This parameter is visible when the parameter "Filter type" is set to "Median" or "Low pass".
[3] This parameter is visible when the parameter "Send value" is set to "On change" or "On change & cyclic".
[4] This parameter is visible when the parameter "Send value" is set to "Cyclic" or "On change & cyclic".
[5] This parameter is visible when the parameter "Additional function" is set to "Send alarm" or "Send bit" or "Send byte" or "Send scene" or "Send percentage". If the low threshold value is higher than the high threshold value and current value is exceed (high threshold) or dropped below (low threshold) the values, just low-level alarm value is sent over "Additional Value" object.
[6] This parameter is visible when the parameter "Additional function" is set to "Send bit" or "Send byte" or "Send scene" or "Send percentage".
[7] This parameter is visible when the parameter "Send bit > Send low-level alarm" is set to "Yes".
[8] This parameter is visible when the parameter "Send bit > Send normal-level alarm" is set to "Yes".
[9] This parameter is visible when the parameter "Send bit > Send high-level alarm" is set to "Yes".

4.5.4. Brightness Internal

This section describes how to configure the parameters for the internal brightness sensor of the iX3. The integrated ambient brightness sensor allows the measuring of the intensity of light value in the room. The brightness sensor can be measured intensity of light up to 3000 Lux.

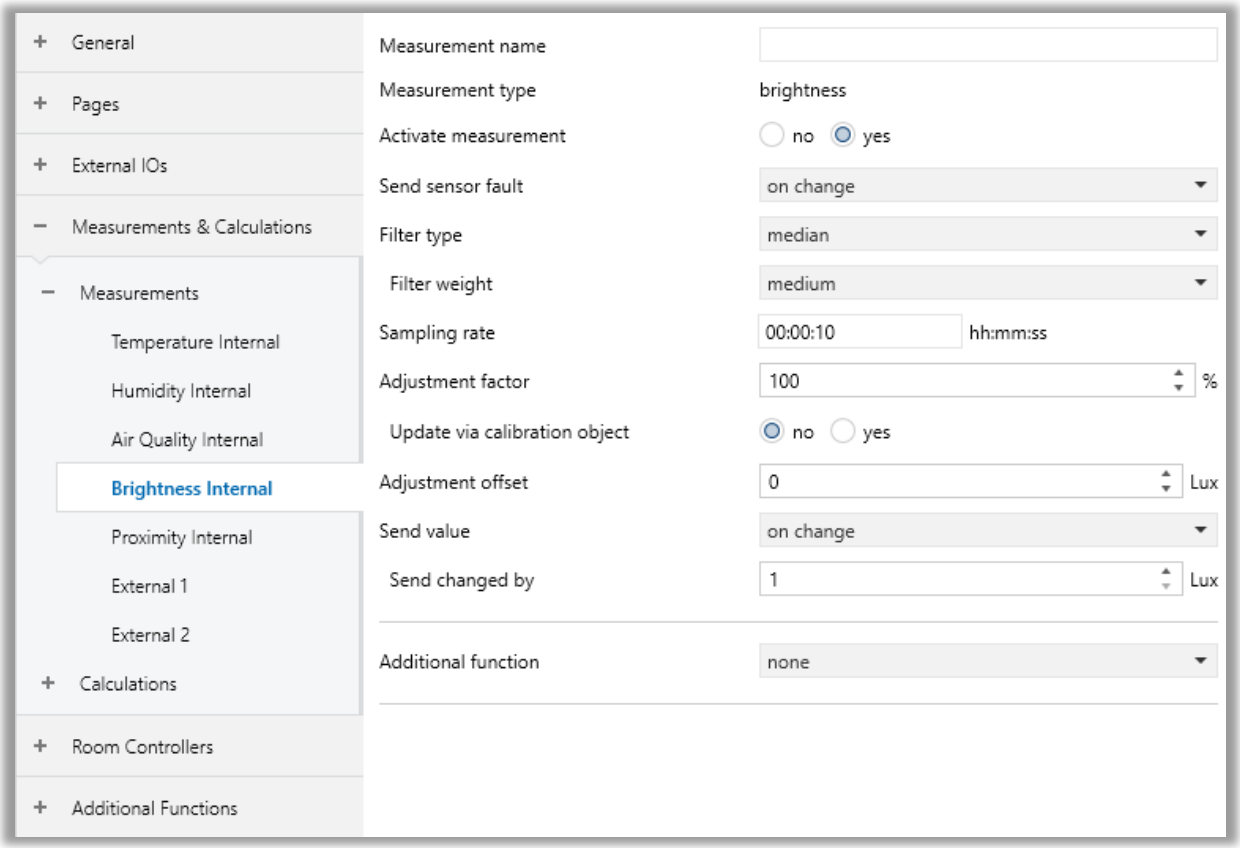


Fig. 35: Brightness Internal Page

Lighting Condition	From (lux)	To (lux)	Average value (lux)
Very dark	11	50	30
Dark Indoors	51	200	125
Dim Indoors	201	400	300
Normal Indoors	401	1000	700
Bright Indoors	1001	5000	3000

Table 6: Examples of Illuminance

4.5.4.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Measurement name	This parameter is used to type a Measurement name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Activate measurement	This parameter is used to enable or disable the measurement.	**No** Yes
Activate measurement: Yes		
Send sensor fault	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission. **Cyclic:** "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	Disable **On change** Cyclic On change & cyclic
-> Send cycle time[1]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams.	00:00:01 … **00:10:00** … 18:12:15
Filter type	This parameter is determined the type of sensor noise filter. **Median**: This filter calculates an average with a series of measured values before sending on the bus. **Low pass**: This filter calculates a value via 1st order IIR filter before sending on the bus.	None **Median** Low pass
-> Filter weight[2]	The parameter is determined the coefficient of the filter. **If median filter is selected;** **Low** = average value every **5** measurements; **Medium** = average value every **15** measurements; **High** = average value every **25** measurements. **If low pass filter is selected;** **Low** = output value relies on new measurement; **Medium** = output value relies on new and previous measurements equally.	Low **Medium** High

	High = output value relies on the previous measurements more	
Sampling rate	The parameter is determined the sampling time of the sensor. For example, sampling rate is selected as 00:00:10, the sensor value is updated per 10 seconds.	00:00:01 … **00:00:10** … 18:12:15
Adjustment factor (%)	This parameter determines the calibration factor. This parameter can be changed on runtime via group object. In this case, the value measured by the sensor is multiplied by 0.01 of the set adjustment factor. Adjustment factor value can be calculated by this formula: Adjustment factor = (The real value that is read from external sensor / device value that is measured internally) × 100	0…**100**...65535
Update via calibration object	If this parameter is set to "**Yes"**, sensor calibration is carried out either via an object.	**No** Yes
Adjustment offset (Lux)	This parameter is used to determine the calibration value of the sensor.	-1200…**0**…1200
Send value	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission. **Cyclic:** "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	Disable **On change** Cyclic On change & cyclic
-> Send changed by (Lux)[3]	This parameter determines the minimum variation for the sensor value to send the object.	**1**…255
-> Send cycle time[4]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams	00:00:01 … **00:00:10** … 18:12:15
Additional function	This parameter is used to determine the additional function of sensor measurement besides sending its value.	**None** Send alarm Send bit

	If "Alarm function" is selected, low-level alarm and high-level alarm can be transmitted to bus via an object. Otherwise, a specific value can be transmitted via object with specific type.	Send byte Send Scene Send Percentage
Low level threshold (Lux)[5]	This parameter determines the low-level value of the additional function. The low threshold must be less than the high threshold.	**0**…1200
High level threshold (Lux)[5]	This parameter determines the high-level value of the additional function. The high threshold must be higher than the low threshold.	0…**500**…1200
Threshold hysteresis (Lux)[5]	This parameter determines the hysteresis value of the additional function.	0…**50**…1200
Send low level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send low level value[7]	The value to be sent when the measurement value is lower than low-level threshold.	Values depend on DPT selection.
Send normal level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send normal level value[8]	The value to be sent when the measurement value is between low-level and high-level threshold.	Values depend on DPT selection.
Send high level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send high level value[9]	The value to be sent when the measurement value is higher than low-level threshold.	Values depend on DPT selection.
Send alarm[5]	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission.	Disable **On change** Cyclic On change & cyclic

	Cyclic: "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	
-> Send cycle time[4]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams	00:00:01 … **00:00:10** … 18:12:15

[1] This parameter is visible when the parameter "Send sensor fault" is set to "Cyclic" or "On change & cyclic" or "Always".
[2] This parameter is visible when the parameter "Filter type" is set to "Median" or "Low pass".
[3] This parameter is visible when the parameter "Send value" is set to "On change" or "On change & cyclic".
[4] This parameter is visible when the parameter "Send value" is set to "Cyclic" or "On change & cyclic".
[5] This parameter is visible when the parameter "Additional function" is set to "Send alarm" or "Send bit" or "Send byte" or "Send scene" or "Send percentage". If the low threshold value is higher than the high threshold value and current value is exceed (high threshold) or dropped below (low threshold) the values, just low-level alarm value is sent over "Additional Value" object.
[6] This parameter is visible when the parameter "Additional function" is set to "Send bit" or "Send byte" or "Send scene" or "Send percentage".
[7] This parameter is visible when the parameter "Send bit > Send low-level alarm" is set to "Yes".
[8] This parameter is visible when the parameter "Send bit > Send normal-level alarm" is set to "Yes".
[9] This parameter is visible when the parameter "Send bit > Send high-level alarm" is set to "Yes".

4.5.6. External X

This section describes how to configure the parameters for the external sensors of the iX3. If external input's type is selected as analog, it is considered as a sensor. Therefore, the end-users can be configured the parameters below measurement channel. Temperature and brightness sensor can be connected to external inputs.

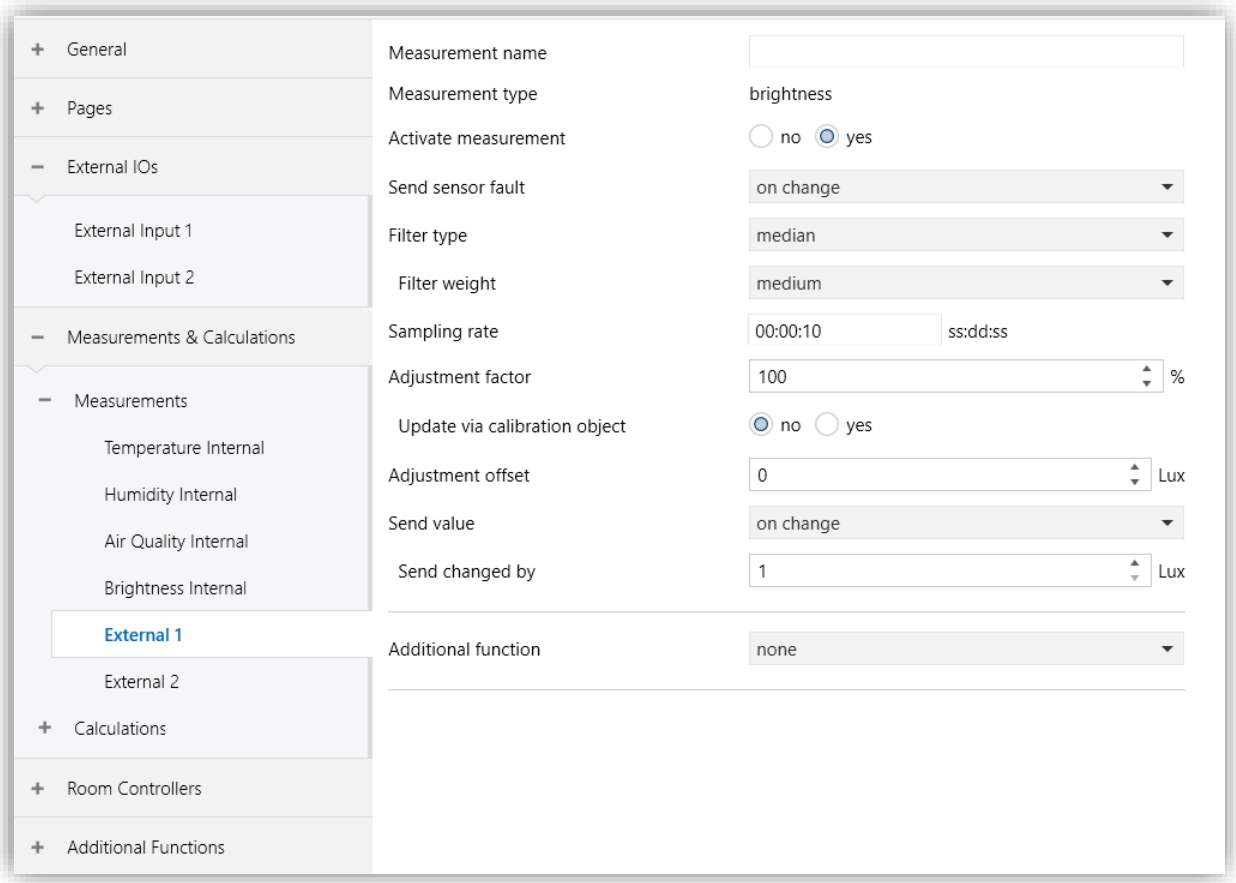


Fig. 37: External X Page

4.5.6.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Measurement name	This parameter is used to type a Measurement name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Activate measurement	This parameter is used to enable or disable the measurement.	**No** Yes
Activate measurement: Yes		
Send sensor fault	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission. **Cyclic:** "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	Disable **On change** Cyclic On change & cyclic
-> Send cycle time[1]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams.	00:00:01 … **00:10:00** … 18:12:15
Filter type	This parameter is determined the type of sensor noise filter. **Median**: This filter calculates an average with a series of measured values before sending on the bus. **Low pass**: This filter calculates a value via 1st order IIR filter before sending on the bus.	None **Median** Low pass
-> Filter weight[2]	The parameter is determined the coefficient of the filter. **If median filter is selected;** **Low** = average value every **5** measurements; **Medium** = average value every **15** measurements; **High** = average value every **25** measurements. **If low pass filter is selected;** **Low** = output value relies on new measurement; **Medium** = output value relies on new and previous measurements equally.	Low **Medium** High

	High = output value relies on the previous measurements more	
Sampling rate	The parameter is determined the sampling time of the sensor. For example, sampling rate is selected as 00:00:10, the sensor value is updated per 10 seconds.	00:00:01 … **00:00:10** … 18:12:15
Adjustment factor	This parameter determines the calibration factor. This parameter can be changed on runtime via group object. In this case, the value measured by the sensor is multiplied by 0.01 of the set adjustment factor. Adjustment factor value can be calculated by this formula: Adjustment factor = (The real value that is read from external sensor / device value that is measured internally) × 100	0…**100**...65535
Update via calibration object	If this parameter is set to "**Yes"**, sensor calibration is carried out either via an object.	**No** Yes
Adjustment offset	This parameter is used to determine the calibration value of the sensor.	-200…**0**…200
Send value	This parameter determines whether and when the value will be sent via an object. **On change:** "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission. **Cyclic:** "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	Disable On change Cyclic On change & cyclic
-> Send changed by[3]	This parameter determines the minimum variation for the sensor value to send the object.	0.1K, 0.2K, 0.3K, 0.5K, 1K, 1.5K, 2K, 2.5K, 3K, 3.5K, 4K, 4.5K, 5K, 7.5K, 10K
-> Send cycle time[4]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams	00:00:01 … **00:00:10** … 18:12:15

Additional function	This parameter is used to determine the additional function of sensor measurement besides sending its value. If "Alarm function" is selected, low-level alarm and high-level alarm can be transmitted to bus via an object. Otherwise, a specific value can be transmitted via object with specific type.	None Send alarm Send bit Send byte Send Scene Send Percentage
Low level threshold[5]	This parameter determines the low-level value of the additional function. The low threshold must be less than the high threshold.	-300…**0**…700
High level threshold[5]	This parameter determines the high-level value of the additional function. The high threshold must be higher than the low threshold.	-300…**0**…700
Threshold hysteresis[5]	This parameter determines the hysteresis value of the additional function.	-200…**0**…200
Send low level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send low level value[7]	The value to be sent when the measurement value is lower than low-level threshold.	Values depend on DPT selection.
Send normal level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send normal level value[8]	The value to be sent when the measurement value is between low-level and high-level threshold.	Values depend on DPT selection.
Send high level alarm[6]	This parameter is available if "Additional function" is set as send 1 bit, scene number, percentage or 1 byte. If this parameter is set to "**Yes**" another parameter will appear so the user can enter the value.	**No** Yes
-> Send high level value[9]	The value to be sent when the measurement value is higher than low-level threshold.	Values depend on DPT selection.
Send alarm[5]	This parameter determines whether and when the value will be sent via an object.	Disable On change

	On change: "On change" means that the value is sent if the measured value has changed by at least the configured value since the last transmission. **Cyclic:** "Cyclic" means that the measured value is transmitted cyclically at the selected time. **On change and cyclic:** The value is sent both on change and cyclic.	Cyclic On change & cyclic
-> Send cycle time[4]	This parameter is visible if the cyclical transmission is active. The send cycle time describes the time used between two cyclically transmitted telegrams	00:00:01 … **00:00:10** … 18:12:15

[1] This parameter is visible when the parameter "Send sensor fault" is set to "Cyclic" or "On change & cyclic" or "Always".
[2] This parameter is visible when the parameter "Filter type" is set to "Median" or "Low pass".
[3] This parameter is visible when the parameter "Send value" is set to "On change" or "On change & cyclic".
[4] This parameter is visible when the parameter "Send value" is set to "Cyclic" or "On change & cyclic".
[5] This parameter is visible when the parameter "Additional function" is set to "Send alarm" or "Send bit" or "Send byte" or "Send scene" or "Send percentage". If the low threshold value is higher than the high threshold value and current value is exceed (high threshold) or dropped below (low threshold) the values, just low-level alarm value is sent over "Additional Value" object.
[6] This parameter is visible when the parameter "Additional function" is set to "Send bit" or "Send byte" or "Send scene" or "Send percentage".
[7] This parameter is visible when the parameter "Send bit > Send low-level alarm" is set to "Yes".
[8] This parameter is visible when the parameter "Send bit > Send normal-level alarm" is set to "Yes".
[9] This parameter is visible when the parameter "Send bit > Send high-level alarm" is set to "Yes".

4.6. Calculations

Calculation functions that are produced the weighted sensor values for special cases. To optimize or correct the sensor regulation in special cases (in large rooms, in presence of strong asymmetry of the sensor distribution, when the installation of the device is in a position not suitable, etc.), the device can use a weighted average between up to 4 sensor values (internal, external 1, external 2 and KNX object). The weighted rates can be in the range of 0 to 255 for each sensor. This means that each input value can be multiplied by in the range of 0 – 255.

Additionally, the alarm function is enabled via a parameter. This feature provides to send alarm value if the calculated value is out of the range of the threshold values.

4.6.1. Calculation X

This section describes how to configure the parameters for the Calculation X channels of the iX3.

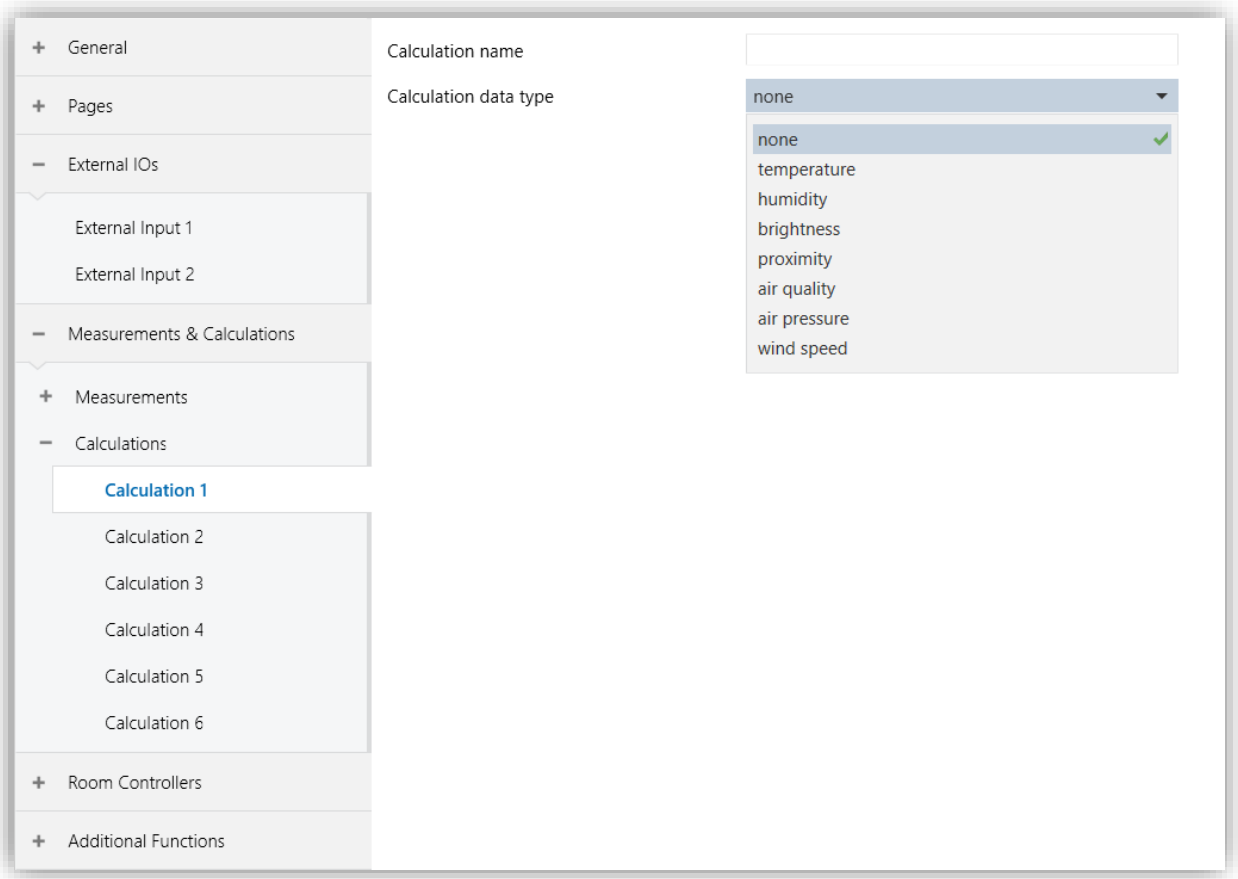


Fig. 38: Calculation X Page

Up to 7 sensor such as temperature, humidity, brightness, proximity, air quality, air pressure, wind speed, can be selected for calculation data type. Each selection has its own parameter configuration screen. Proximity, air pressure and wind speed data type can be used via just "KNX probe" source.

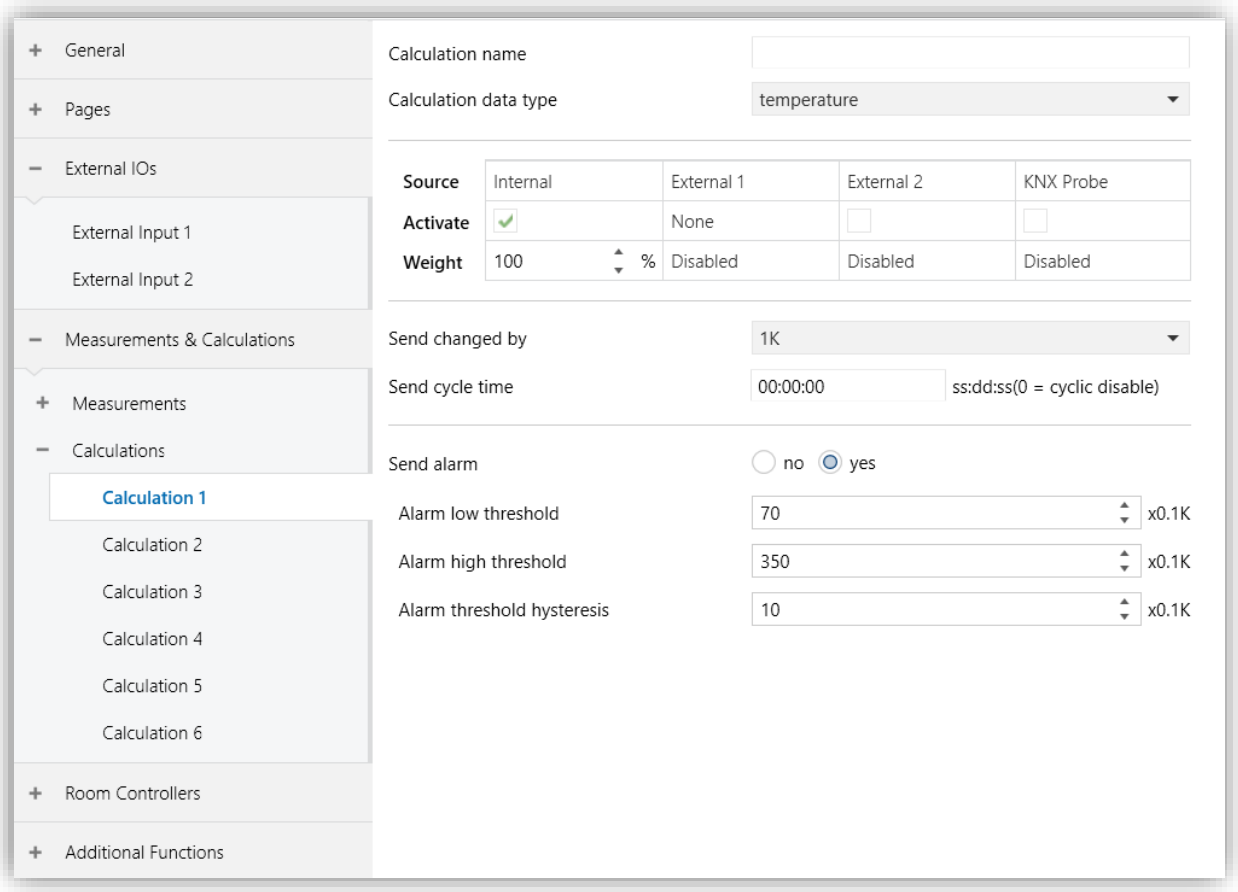


Fig. 39: Calculation for Temperature Page

The calculated value can periodically be sent on the bus with a specified transmission interval, and whenever a specified variation occurs. If KNX probe is selected as source, "KNX probe calibration offset" and "KNX probe surveillance time" parameter is visible. "KNX probe calibration offset" is used to calibrate the received value. "KNX probe surveillance time" parameter is used to determine the surveillance time for the KNX probe. If KNX probe value can't be received per set time, an alarm object is sent to warn the source device.

To optimize or correct the sensor regulation in special cases (in large rooms, in presence of strong asymmetry of the sensor distribution, when the installation of the device is in a position not suitable, etc.), the device can then use a weighted average between up to 4 sensor values (internal, external 1, external 2 and KNX object).

4.6.1.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Calculation name	This parameter is used to type a Led name. The name can be consisting of 40 characters.	**40 bytes allowed**
Calculation data type	This parameter is used to determine the data type to be taken into account for calculation.	**None** Temperature Humidity Brightness Proximity Air quality Air pressure Wind speed
Internal Activate	This parameter is used to determine the activation of internal sensor for calculation. If this parameter is checked, the internal sensor is taken into account for calculation.	**Checked** Unchecked
-> Internal Weight[1]	This parameter is used to determine the weight coefficient of internal sensor for calculation.	0…**100**…255
External X Activate	This parameter is used to determine the activation of external input for calculation. If this parameter is checked, the external input is taken into account for calculation.	**Checked** Unchecked
-> External X Weight[2]	This parameter is used to determine the weight coefficient of external input for calculation.	0…**100**…255
KNX Probe Activate	This parameter is used to determine the activation of KNX probe object for calculation. If this parameter is checked, the KNX probe value is taken into account for calculation.	**Checked** Unchecked
-> KNX Probe Weight[3]	This parameter is used to determine the weight coefficient of KNX probe value for calculation.	0…**100**…255
-> KNX probe calibration offset[3]	This parameter is used to determine the calibration value is received from the KNX Probe temperature object.	Values depend on DPT selection.
->KNX probe surveillance time[3]	This parameter is used to determine the surveillance time for the KNX probe. If this parameter is	**0**…255

	configured higher than 0, "Probe Surveillance" object will be visible. E.g., if this parameter is configured as 10. Every 10 min the received value from KNX is taken into account for calculation.	
Send changed by	This parameter determines the minimum variation value for the output of calculation object to send a value.	Values depend on DPT selection.
Send cycle time	This parameter determines the time of control value to be sent periodically.	00:00:01 … **00:00:00** … 18:12:15
Send alarm	This parameter is used to enable the alarm objects to define a threshold value for alarm information.	**No** Yes
-> Alarm low threshold[4]	This parameter determines the calculation object's low threshold value.	Values depend on DPT selection.
-> Alarm high threshold[4]	This parameter determines the calculation object's high threshold value.	Values depend on DPT selection.

[1] This parameter is visible when the parameter "Internal Activate" is set to "Checked".
[2] This parameter is visible when the parameter "External X Activate" is set to "Checked".
[3] This parameter is visible when the parameter "KNX Probe Activate" is set to "Checked".
[4] This parameter is visible when the parameter "Send alarm" is set to "Yes".

4.7. Room Controller - Thermostat

All configurations related to thermostat control on the iX3 are described in the sections of this chapter. This parameter page will be shown when it is enabled in the "General" parameter page section. The information about the "General" parameter configuration section is described after the theoretical control type expressions that are given below.

- 2 points/Proportional fan controller that can be used by main and additional heating/cooling systems.
- Thermostat weekly program.
- Energy saving function for thermostat functions.
- Temperature limitation for thermostat functions.

4.7.1. Control Types Theoretical Explanations

The room controller device can be used for only heating, only cooling or heating and cooling. If the room controller is in heating and cooling mode, the transition from heating to cooling or vice versa can occur automatically. The thermostat measures the actual temperature of the ambient air and continuously compares it to the set temperature, and the controller automatically calculates whether to send a control signal for heating or cooling.

The control algorithm based on the difference between the desired setpoint temperature values and the measured actual temperature values processes a command value that can be either percentage or ON / OFF. The command, periodically or depending on the event, is transmitted to a KNX actuator device via a bus line with communication objects.

4.7.1.1. 2-Points Control

This control algorithm, also known as ON / OFF, is the most classic and popular one. The algorithm follows a hysteresis cycle, allowing the system to switch ON / OFF. Hence, 2 switching levels are considered for switching.

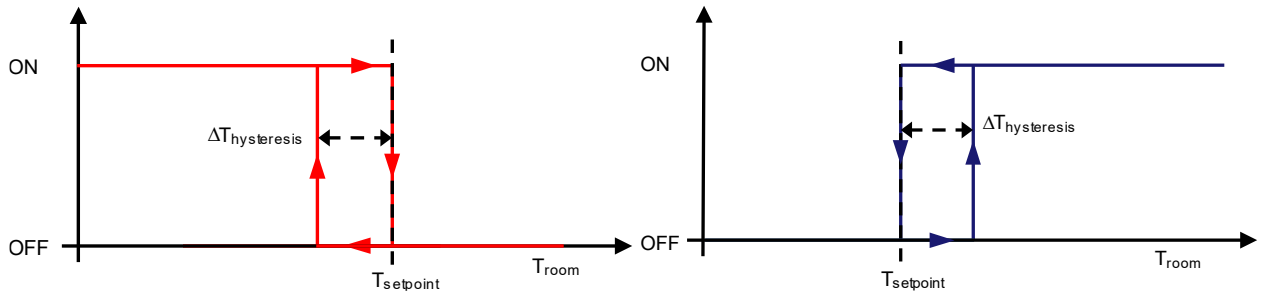


Fig. 40: 2 – Points Control Hysteresis Cycle

Heating mode

When the measured temperature is lower than the difference between the setpoint and the hysteresis value ($T_{setpoint} - \Delta T_{hysteresis}$), the device activates the heating system by sending the KNX command to the actuator that controls the heating system via connected to a related group address. When the measured temperature reaches the setpoint temperature, the device sends a related command and deactivates the heating system. In this way, there are 2 decision thresholds to activate and deactivate the heating system. The first one is the temperature at which the device activates the system ($T_{setpoint} - \Delta T_{hysteresis}$), and the second one is the temperature at which the device deactivates the heating system ($T_{setpoint}$).

Cooling mode

When the measured temperature is higher than the difference between the setpoint and the hysteresis value ($T_{setpoint} - \Delta T_{hysteresis}$), the device activates the heating system by sending the KNX command to the actuator that controls the cooling system via connected to a related group address. When the measured temperature reaches the setpoint temperature, the device sends a related command and deactivates the cooling system. In this way, there are 2 decision thresholds to activate and deactivate the cooling system. The first one is the temperature at which the device activates the system ($T_{setpoint} + \Delta T_{hysteresis}$), and the second one is the temperature at which the device deactivates the heating system ($T_{setpoint}$). There are 2 different parameters for heating and cooling hysteresis values in the ETS programme. Values differ depending on the system type.

4.7.1.2. Continuous (PI) Control

Proportional – Integral control (PI control) is explained by the relationship shown below:

$$\boldsymbol{control\ variable}(\boldsymbol{t}) = \boldsymbol{Kp} \times \boldsymbol{error}(\boldsymbol{t}) + \boldsymbol{Ki} \times \int_0^t \boldsymbol{error}(\boldsymbol{t})\, \boldsymbol{dt}$$

whereby:

$$error(t) = (Setpoint - Measured\ temperature)\ in\ heating$$
$$error(t) = (Measured\ temperature - Setpoint)\ in\ cooling$$
$$Kp = proportional\ constant$$
$$Ki = integral\ constant$$

The control variable contains integral and proportional (Ki and Kp) constants to eliminate errors. In practice, intuitively generated values are generally used.

Ex 1:

$$\boldsymbol{Proportional\ band\ BP}\ [\boldsymbol{K}] = \boldsymbol{100 / Kp} \qquad \boldsymbol{Integral\ time\ Ti}\ [\boldsymbol{min}] = \boldsymbol{Kp / Ki}$$

The proportional band is the error value that determines the maximum deflection output as 100%.
For example, a regulator with a proportional band of 5 K provides a 100% control output when the Setpoint = 20°C and the measured temperature is ≤ 15°C in heating; in the cooling conduction mode, it provides a 100% control output when the Setpoint = 24°C and the measured temperature is ≥ 29°C. As shown in the figure, a regulator with a small proportional band tends to provide higher values of the control variable for small errors than a regulator with a higher proportional band.

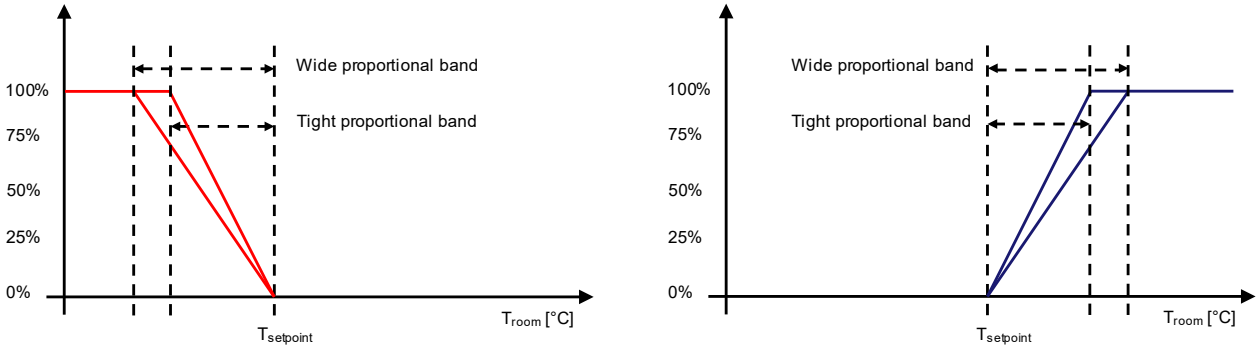


Fig. 41: Continuous PI Control Proportional Band Widths

The integral time is the time required to repeat the value of the control variable of a purely proportional regulator when the error remains constant in time.

Ex 2:
For example, with a purely proportional controller in heating and with a value of proportional band of 4 K, if the setpoint is = 20°C and the measured temperature is = 18°C, the control variable assumes the value of 50%. With an integral time = 60 minutes, if the error remains constant, the control variable will take the value = 100% after 1 hour, i.e., a contribution equal to the value given by only proportional contribution will be added to the control variable. In heating and air conditioning systems, a purely proportional controller is not able to guarantee the achievement of the setpoint. You should always introduce an integrated action for achieving the Setpoint: that is why the integral action is also called automatic reset.

4.7.1.3. PWM (PI) Control

The PWM (Pulse Width Modulation) proportional-integral controller allows the digital output to be set to ON and OFF by sampling an analogue control variable within a specified period. The controller runs periodically through a cycle and keeps its output ON for each period in proportion to the value of the control variable. As shown in the below figure, by varying the ratio between the "ON" time and the "OFF" time, the average activation time of the output changes, and as a result, the average heating or cooling power supplied by the room changes.

The cycle time for the control value for the PWM signal calculated from the PI controller's control value is specified. Depending on the control value, the selected cycle time is divided into an ON and OFF signal. Therefore, a control value output of 50 % with a PWM cycle of 12 min signifies an ON phase of 6 min. and an OFF phase of 6 min.

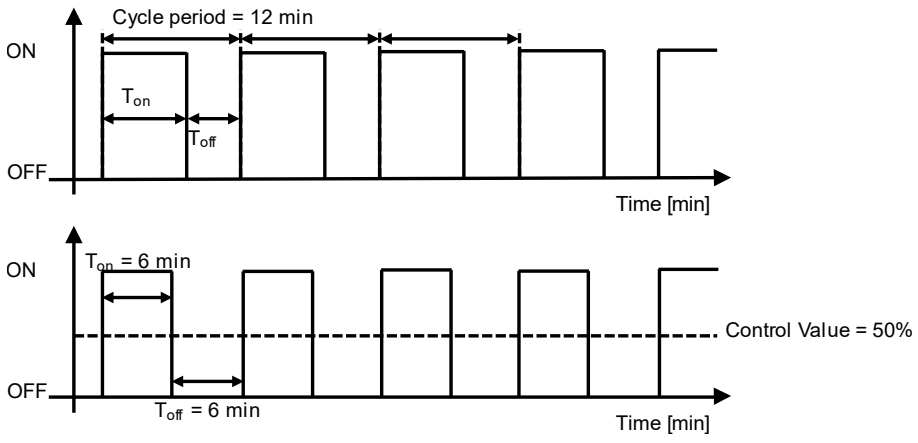


Fig. 42: PWM Control Sampling

This type of control is well suited for use with ON / OFF actuators, such as electrothermal actuators and drives for zone valves, which are less expensive than proportional actuators.

A distinctive advantage of this type of control is that it eliminates the inertia of the system: it allows significant energy savings because unnecessary interventions on the system introduced by the 2-point control with hysteresis are avoided and only the power is required to compensate for the losses.

Every time the changes the desired temperature setpoint is, the cycle time is interrupted, the control output is reprocessed and the PWM restarts with a new cycle: this allows the system to reach its steady state more quickly.

Terminal Type	Proportional Ban [K]	Integral Time [min]	Cycle Period [min]
Radiators	5	150	15-20
Electrical heaters	4	100	15-20
Fan-coil	4	90	15-20
Floor radiant panels	5	240	15-20
Ceiling radiant panels	5	100	15-20

Table 7: Guidelines for choosing the proper parameters of a PMW PI controller

Guidelines for choosing the proper parameters of a PMW Proportional-Integral controller:

- Cycle time: for low-inertial systems such as heating and air conditioning systems, short cycle times must be chosen (10-15 minutes) to avoid oscillations of the room temperature.
- Narrow proportional band: wide and continuous oscillations of the room temperature, short setpoint settling time.
- Wide proportional band: small or no oscillations of the room temperature, long setpoint settling time.
- Short integral time: short setpoint settling time, continuous oscillations of the room temperature.
- Long integral time: long setpoint settling time, no oscillations of the room temperature.

4.7.2. Thermostat X

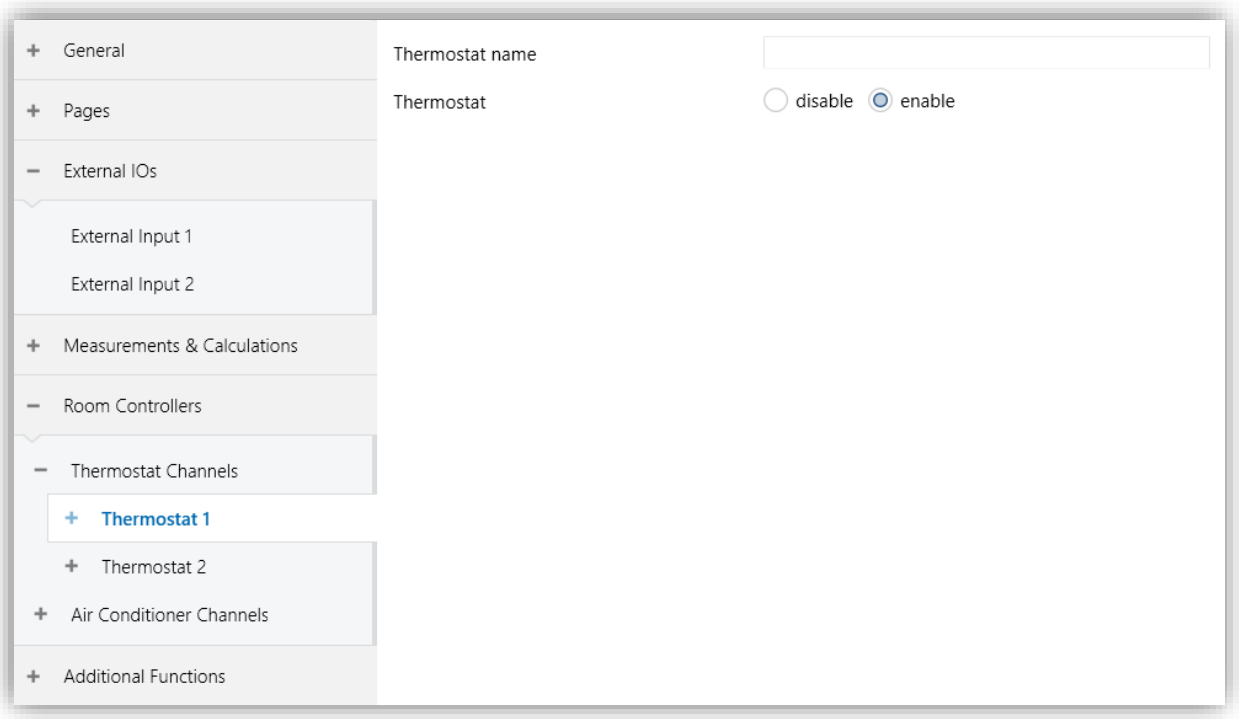


Fig. 43: Room Controller Thermostat Configuration Section

4.7.2.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Thermostat name	This parameter is used to type a Thermostat name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Thermostat	This parameter is used to control the thermostat features.	**Disable** Enable

4.7.3. Thermostat - General

The thermostat function can be selected as the "master" controller or "slave" controller in the configuration settings in this section. When the selection is made as to the "master" controller, configuration sections and the communication objects are opened to define the thermostat functions. When the selection is made as to the "slave" controller, some configuration sections related to the thermostat functions are disabled. The slave controller must be connected to the master controller with the KNX communication object as it will operate as a dependent controller with commutations object. In thermostat slave mode, setpoint adjustment, thermostat activation control, heating/cooling switchover and operation mode control can be made. Also, LCD can be used as fan indicator in slave operation or fan controller isn't used for thermostat.

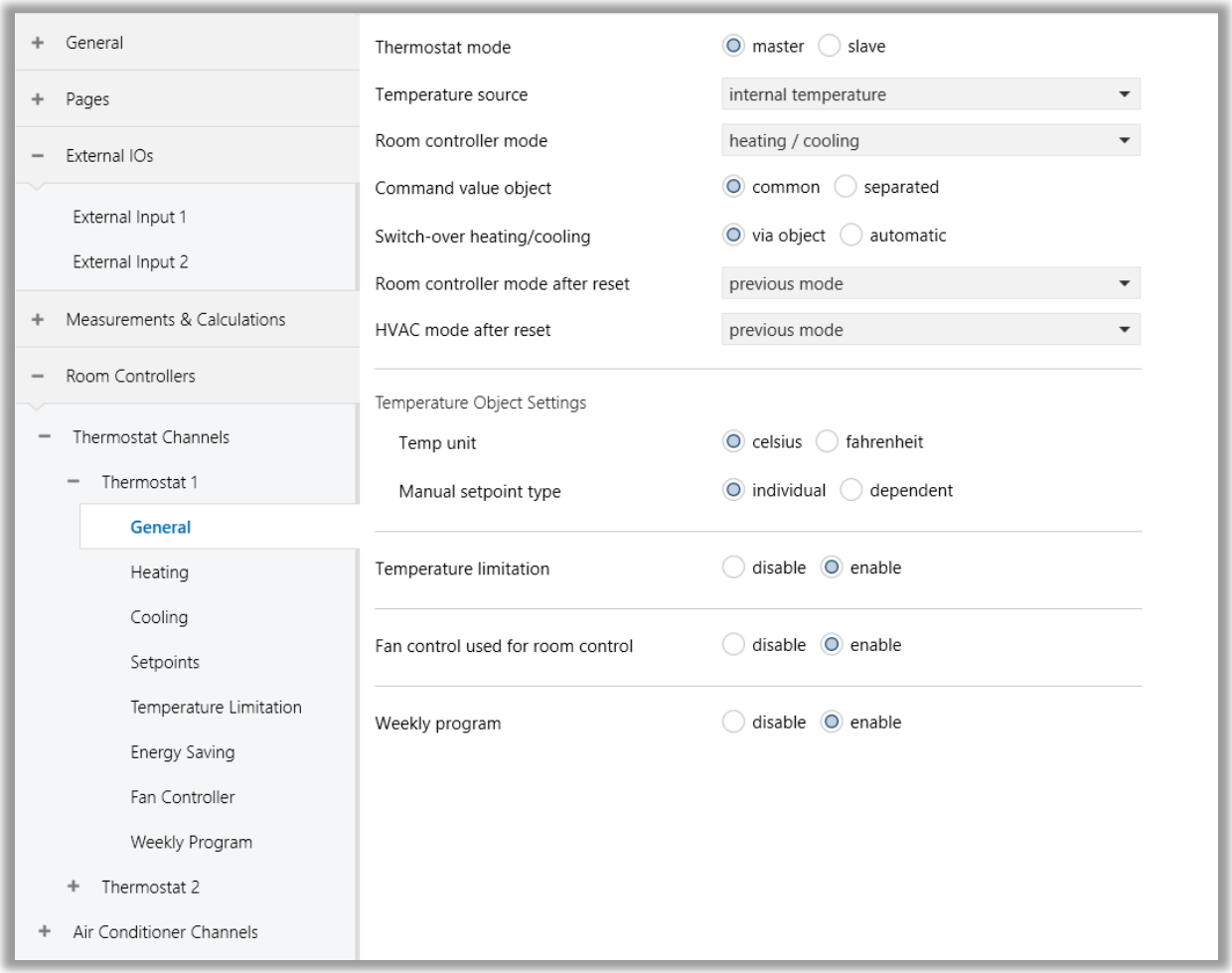


Fig. 44: Room Controller Thermostat General Configuration Section

4.7.3.1. Parameters List

PARAMETER	DESCRIPTION	VALUES
Thermostat mode	The thermostat function's operating type is determined with this parameter.	**Master** Slave
Temperature source	This parameter determines the temperature source of room controller. If thermostat temperature value is higher/lower than the setpoints of the protection mode's setpoint max/min limit values, the active operation mode is changed as Protection mode. After that the end-users can change the operation mode again.	**Internal temperature** Temperature object Calculation 1…6
Room controller mode[1]	Room controller mode is determined with this parameter.	**Heating** Cooling Heating & Cooling
HVAC mode after reset[1]	This parameter determines the operating mode of the room controller after a reset occurs. **Ex:** When a power failure occurs.	**Previous value** Comfort Standby Economy Protection
Command value object[2]	The object types of temperature command values for heating and cooling mode are determined with this parameter.	**Common** Separated
Switch-over heating / cooling[2]	This parameter determines how the heating/cooling transition is made. If heating/cooling switch-over mode isn't Automatic, the user can be configured heating or cooling setpoint. If heating/cooling switch-over mode is Automatic, the user can't be configured that the cooling setpoint is higher than the heating setpoint. In automatic mode the cooling setpoint is equal the heating setpoint at least. If an input value that is higher than heating setpoint, is received over "Cooling [Operation Mode] Setpoint Temperature" object, received telegram is ignored.	**Via communication object** Automatic

Room controller mode after reset[3]	This parameter determines the room controller mode of the room controller after a reset occurs. **Ex:** When a power failure occurs.	Heating Cooling **Previous mode**
Temp Unit	The temperature unit type to be used by thermostat objects is defined by this parameter.	**Celsius** Fahrenheit
Manual setpoint type	The desired temperature value can be controlled with individual or dependent setpoints by this parameter. **Individual setpoint:** The input value must be the desired setpoint. **Dependent setpoint:** The input value must be the difference of desired setpoint according to base setpoint.	**Individual** Dependent
Temperature limitation	This parameter enables temperature limitation function of thermostat.	**Disable** Enable
Fan control used for room control[1]	This parameter determines the fan controls that are used inside or outside of the thermostat function. If the it is selected to use outside of the thermostat function, just the fan states will be displayed on the device as fan indicators.	**Disable** Enable
Weekly program	This parameter enables weekly program of thermostat.	**Disable** Enable

[1] This parameter is visible when the parameter "Thermostat mode" is set to "Master".
[2] This parameter is visible when the parameter "Room controller mode" is set to "Heating / cooling".
[3] This parameter is visible when the parameter "Switch-over heating/cooling" is set to "Via object".

4.7.4. Thermostat - Heating

The device's operation principle of the heating feature is as follows: When the measured temperature is lower than the setpoint temperature, the device activates the heating system by sending a KNX command to the actuator that controls the heating system via connected to the related group address. When the measured temperature reaches the setpoint temperature, the device sends a related command and deactivates the heating system. The heating feature can be controlled with different types of configuration settings. These configuration settings are as follows;

Selection of the "Heating 2 – Points Control" parameter, 1-bit / 1-byte on/off control can be selected.

Selection of the "Heating PWM Control" parameter, 1-bit / 1-byte on/off control can be selected.

Selection of the "Heating Continuous Control" parameter, 1-byte proportional-integral control.

4.7.4.1. Heating 2 – Points Control

When the measured temperature is lower than the difference between the setpoint and the hysteresis value ($T_{setpoint} - \Delta T_{hysteresis}$), the device activates the heating system by sending a KNX command to the actuator that controls the heating system via connected to a related group address. When the measured temperature reaches the setpoint temperature, the device sends a related command and deactivates the heating system. In this way, there are 2 decision thresholds to activate and deactivate the heating system. The first one is the temperature at which the device activates the system ($T_{setpoint} - \Delta T_{hysteresis}$), and the second one is the temperature at which the device deactivates the heating system ($T_{setpoint}$).

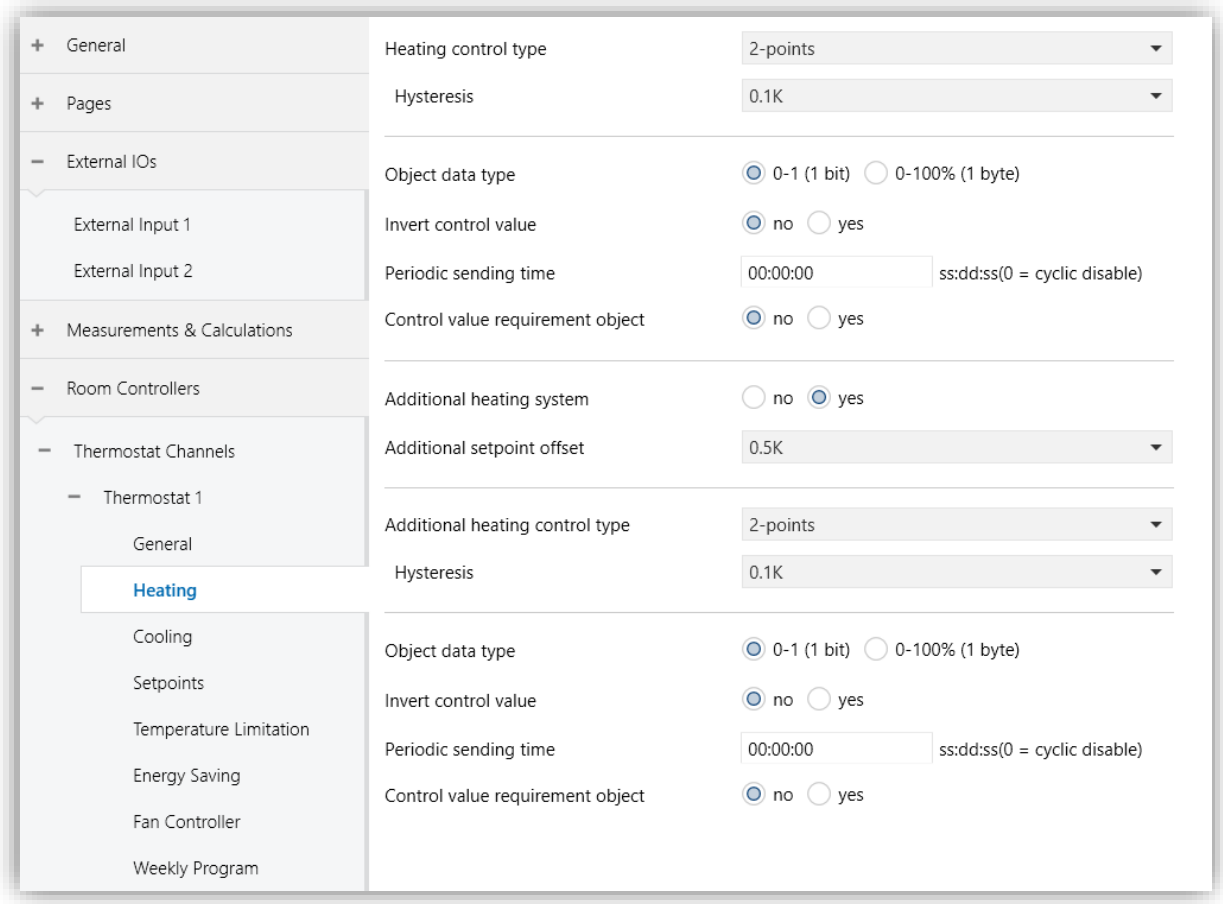


Fig. 45: Heating 2-Points Control Configuration

4.7.4.2. Parameters List

PARAMETER	DESCRIPTION	VALUES
Heating control type	This parameter determines the heating control type.	**2 – points** PWM Continuous
Hysteresis	This parameter determines the hysteresis value.	0.1K…2.0K
Object data type	This parameter is used to determine data type of control value object.	**0-1 (1 bit)** 0-100% (1 byte)
Invert control value	This parameter is used to invert control output.	**No** Yes
Periodic sending time	This parameter is used to periodically send the commands to the bus line.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the heating system.	**No** Yes
Additional heating system	This parameter activates the additional heating system.	**No** Yes

4.7.4.3. Heating PWM Control

The PWM (Pulse Width Modulation) proportional-integral controller allows the digital output to be set to ON and OFF by sampling an analogue control variable within a specified period. The controller runs periodically through a cycle and keeps its output ON for each period in proportion to the value of the control variable. By varying the ratio between the "ON" time and the "OFF" time of the heating system, the average activation time of the output changes, and as a result, the average heating power supplied by the room changes.

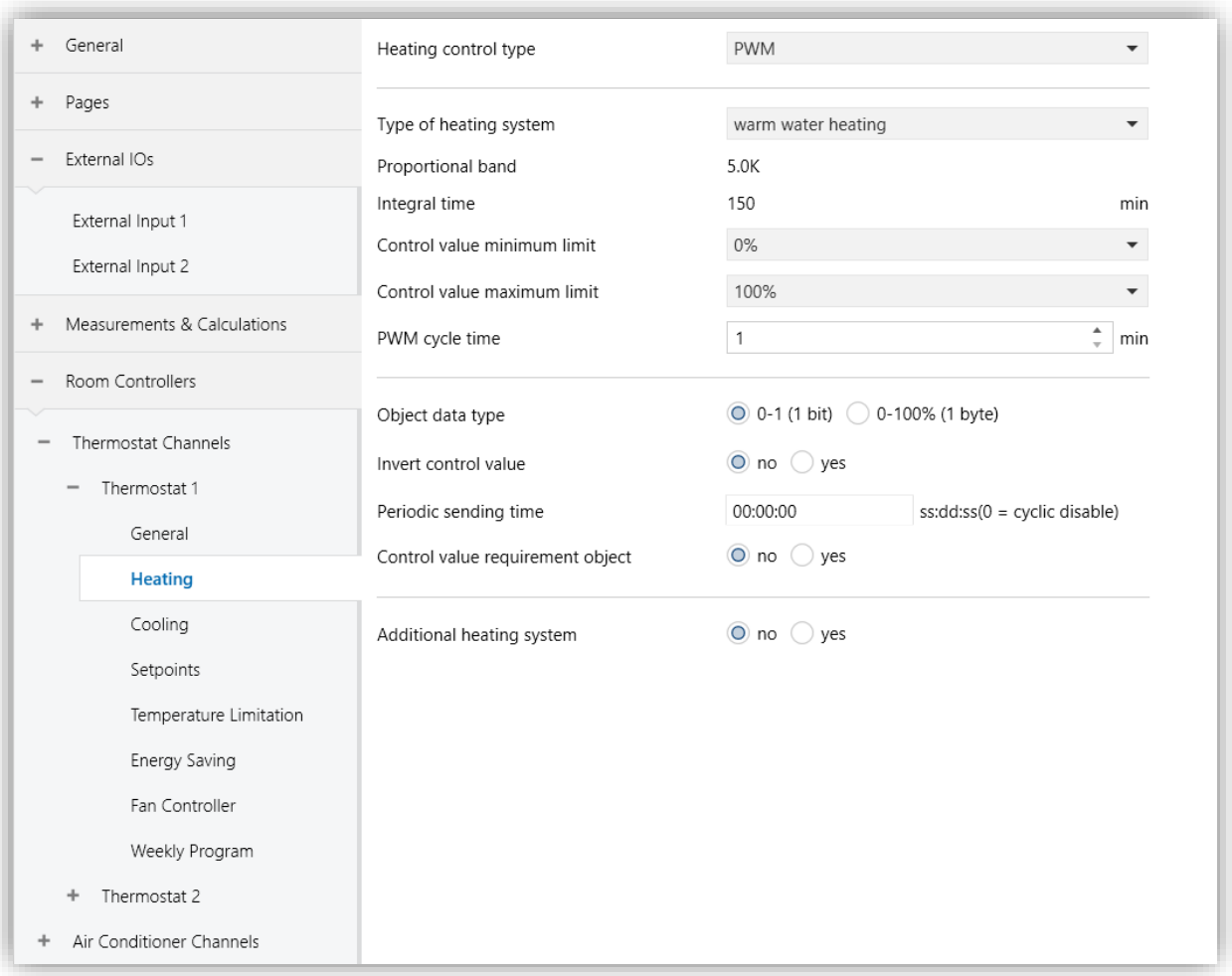


Fig. 46: Heating PWM Control Configuration

4.7.4.4. Parameters List

PARAMETER	DESCRIPTION	VALUES
Type of heating system	This parameter determines the heating system to be controlled.	**Warm water heating** Electric heating Floor heating Split unit Fan coil User defined
Proportional band (K)	This parameter determines the proportional band.	**5.0K** (0.5K…10.0K)
Integral time (min)	This parameter determines the integral time.	**150** (0…255)
Control value minimum (%)	This parameter determines the output object's minimum control value.	**0%** (0%, 5%, 10%, 15%, 20%, 25%, 30%)
Control value maximum (%)	This parameter determines the output object's maximum control value.	**100%** (70%, 75%, 80%, 85%, 90%, %95, 100%)
PWM cycle time (min)	This parameter determines the PWM cycle time.	**1**…255
Object data type	This parameter is used to determine data type of control value object.	**0-1 (1 bit)** 0-100% (1 byte)
Invert control value	This parameter is used to invert control output.	**No** Yes
Periodic sending time	This parameter is used to periodically send the commands to the bus line.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the heating system.	**No** Yes

4.7.4.5. Heating Continuous Control

Proportional – Integral control (PI control) is explained by the relationship shown below:

$$\boldsymbol{control\ variable(t)\ =\ Kp\ \times\ error(t)\ +\ Ki\ \times \int_0^t error(t)\ dt}$$

whereby:

$$error(t)\ =\ (Setpoint\ -Measured\ temperature)\ in\ heating$$
$$error(t)\ =\ (Measured\ temperature\ -Setpoint)\ in\ cooling$$
$$Kp\ =\ proportional\ constan$$
$$Ki\ =\ integral\ constant$$

The control variable contains integral and proportional (Ki and Kp) constants to eliminate errors. In practice, intuitively generated values are generally used.

Ex 1:

$$\boldsymbol{Proportional\ band\ BP\ [K] = \frac{100}{Kp}}$$

$$\boldsymbol{Integral\ time\ Ti\ [min]\ =\ Kp\ /\ Ki}$$

The proportional band is the error value that determines the maximum deflection output as 100%.

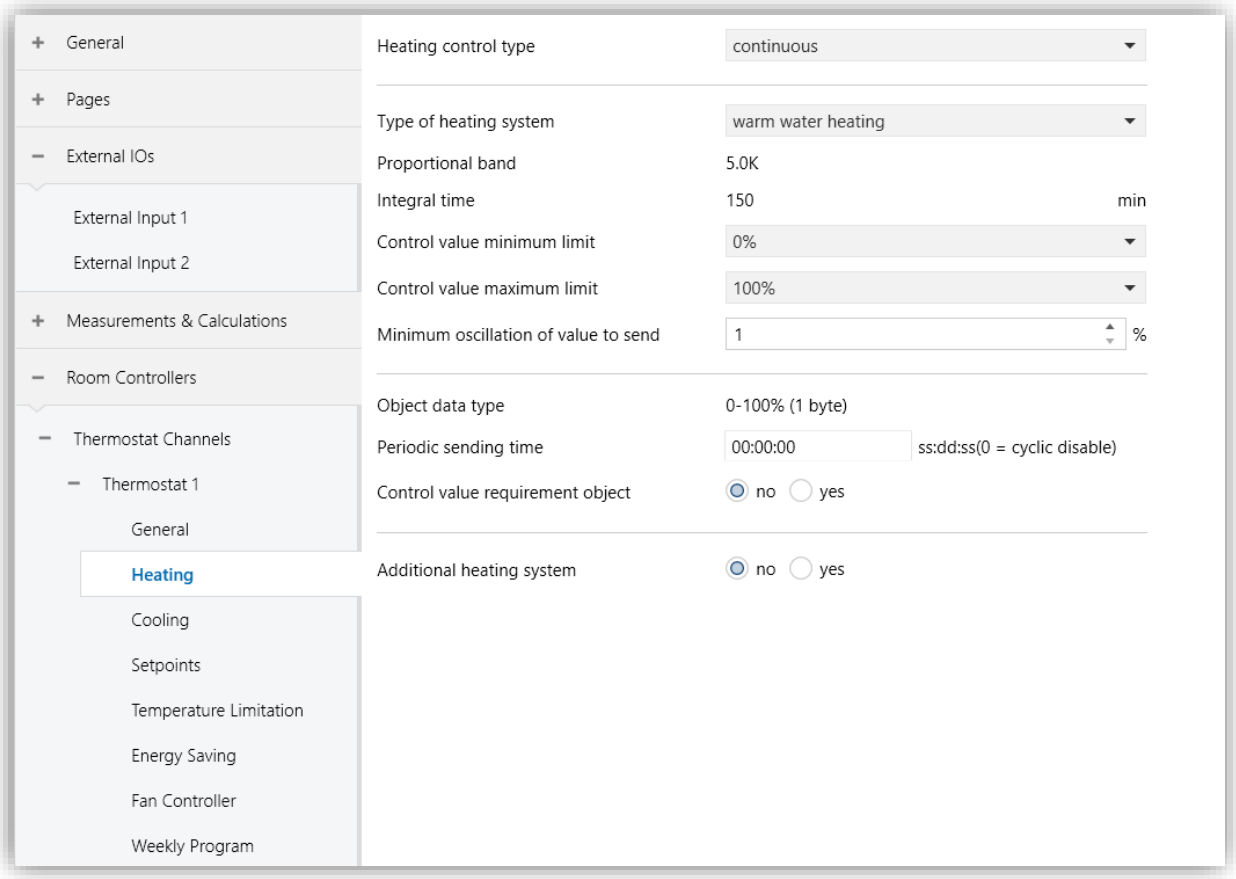


Fig. 47: Heating Continuous Control Configuration

4.7.4.6. Parameters List

PARAMETER	DESCRIPTION	VALUES
Type of heating system	This parameter determines the heating system to be controlled.	**Warm water heating** Electric heating Floor heating Split unit Fan coil User defined
Proportional band (K)	This parameter determines the proportional band.	**5.0K** (0.5K … 10.0K)
Integral time (min)	This parameter determines the integral time.	**150** (0 … 255)
Control value minimum (%)	This parameter determines the output object's minimum control value.	**0%** (0%, 5%, 10%, 15%, 20%, 25%, 30%)
Control value maximum (%)	This parameter determines the output object's maximum control value.	**100%** (70%, 75%, 80%, 85%, 90%, 95%, 100%)
Minimum oscillation of value to send (%)	This parameter determines the minimum oscillation value for the output object to send a value.	**3** (0…100)
Periodic sending time	This parameter is used to periodically send the commands to the bus line.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the heating system.	**No** Yes

4.7.3.7. Additional Heating System

All types of heating controls (2-points, PWM and continuous control) have additional heating system options. The additional heating system works in all control types with the same characteristics. The system activates itself according to the offset configuration. If ($T_{setpoint} - \Delta T_{offset}$) is lower than the ambient room temperature, the additional heating system will be activated according to controller type.

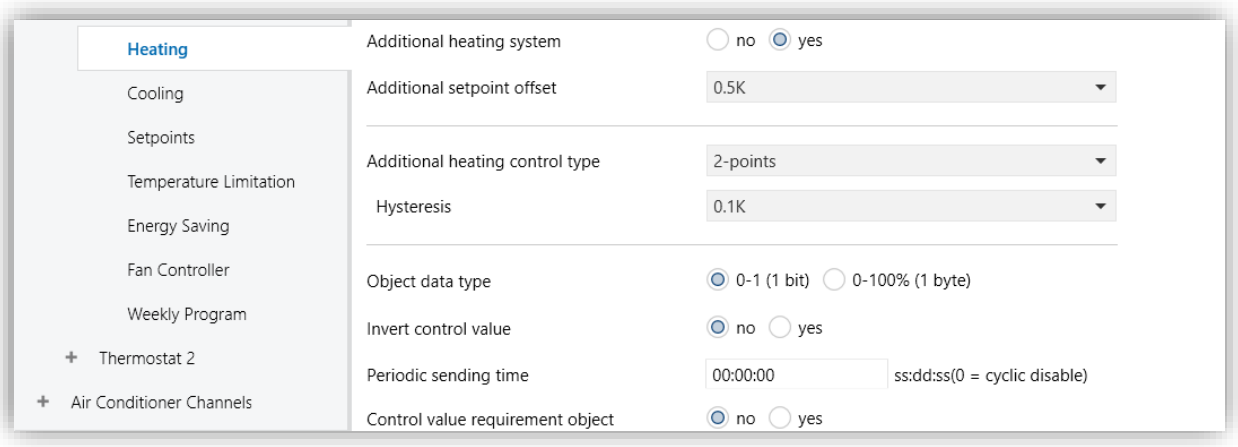


Fig. 48: Additional Heating System Configuration

In additional heating control, 2 - Points and PI Continuous controller heat the room until the difference between ($T_{setpoint} - T_{room}$) is equal to "Additional setpoint offset" parameter.

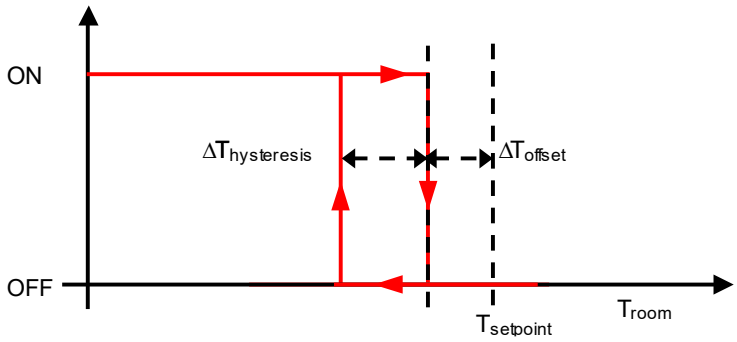


Fig. 49: 2 – Points Hysteresis Cycle for Additional Heating Control

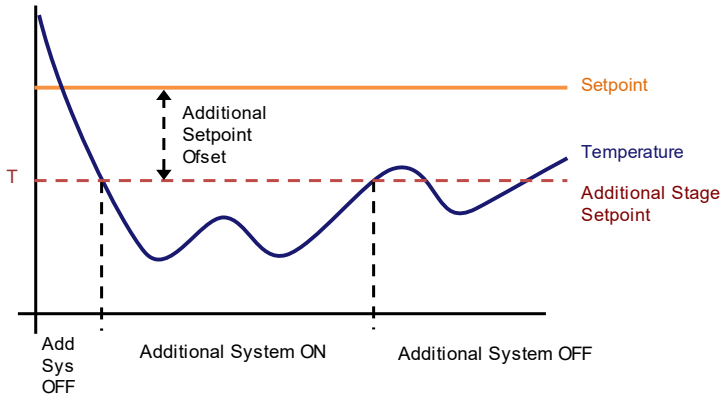


Fig. 50: PI Continuous Graph for Additional Heating Control

4.7.3.7. Parameters List

PARAMETER	DESCRIPTION	VALUES
Additional heating system	This parameter activates the additional heating system.	**No** Yes
Additional setpoint offset	This parameter determines the difference between the setpoint temperature value and the additional heating system's setpoint temperature value.	**0.5K** … 5.0K (°C) **0.9K** … 9.0K (°F)
Additional heating control type	This parameter determines the additional heating system's control object type.	**2 – points** PWM Continuous
Additional heating control type: 2-points		
Hysteresis Value	This parameter determines the hysteresis value.	**0.1K**…2.0K (°C) **0.18K**…3.6K (°F)
Object type	This parameter is used to determine the data type of the control value object.	**0-1 (1 bit)** 0-100% (1 byte)
Invert control value	This parameter is used to invert control output.	**No** Yes
Periodic sending time	This parameter determines the time of control value to be sent periodically.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the additional heating system.	**No** Yes
Additional heating control type: PWM		
Type of additional heating system	This parameter determines the heating system to be controlled.	**Warm water heating** Electric heating Floor heating Split unit Fan coil User defined
Proportional band	This parameter determines the proportional band.	0.5K… **5.0K** … 10.0K (°C) 0.9K… **9.0K** … 18.0K (°F)

Integral time	This parameter determines the integral time.	0 … **90** … 255
Control value minimum limit	This parameter determines the output object's minimum control value.	**0%**, 5%, 10%, 15%, 20%, 25%, 30%)
Control value maximum limit	This parameter determines the output object's maximum control value.	70%, 75%, 80%, 85%, 90%, %95, **100%**
PWM cycle time (min)	This parameter determines the PWM cycle time.	**1**…255
Object data type	This parameter is used to determine data type of control value object.	**0-1 (1 bit)** 0-100% (1 byte)
Invert control value	This parameter is used to invert control output.	**No** Yes
Periodic sending time	This parameter is used to periodically send the commands to the bus line.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the additional heating system.	**No** Yes
Additional heating control type: Continuous		
Type of additional heating system	This parameter determines the heating system to be controlled.	Warm water heating Electric heating Floor heating Split unit **Fan coil** User defined
Proportional band	This parameter determines the proportional band.	0.5K… **5.0K** … 10.0K (°C) 0.9K… **9.0K** … 18.0K (°F)
Integral time	This parameter determines the integral time.	0 … **90** … 255
Control value minimum limit	This parameter determines the output object's minimum control value.	**0%** (0%, 5%, 10%, 15%, 20%, 25%, 30%)
Control value maximum limit	This parameter determines the output object's maximum control value.	**100%** (70%, 75%, 80%, 85%, 90%, %95, 100%)
Minimum oscillation of value to send	This parameter determines the minimum oscillation value for the output object to send a value.	**1** … 100
Periodic sending time	This parameter is used to periodically send the commands to the bus line.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the additional heating system.	**No** Yes

4.7.5. Thermostat - Cooling

The device's operation principle of cooling feature is as follows: When the measured temperature is higher than the setpoint temperature, the device activates the cooling system by sending a KNX command to the actuator that controls the cooling system via connection to the related group address. When the measured temperature reaches the setpoint temperature, the device sends a related command and deactivates the cooling system. The cooling feature can be controlled with different types of configuration settings. These configuration settings are as follows;

Selection of the "Cooling 2 – Points Control" parameter, 1-bit / 1-byte on/off control can be selected.

Selection of the "Cooling PWM Control" parameter, 1-bit / 1-byte on/off control can be selected.

Selection of the "Cooling Continuous Control" parameter, 1-byte proportional-integral control.

4.7.5.1. Cooling 2 – Points Control

When the measured temperature is higher than the difference between the setpoint and the hysteresis value ($T_{setpoint} + \Delta T_{hysteresis}$), the device activates the cooling system by sending a KNX command to the actuator that controls the cooling system via connected to a related group address. When the measured temperature reaches the setpoint temperature, the device sends a related command and deactivates the cooling system. In this way, there are 2 decision thresholds to activate and deactivate the cooling system. The first one is the temperature at which the device activates the cooling system ($T_{setpoint} + \Delta T_{hysteresis}$), and the second one is the temperature at which the device deactivates the cooling system ($T_{setpoint}$).

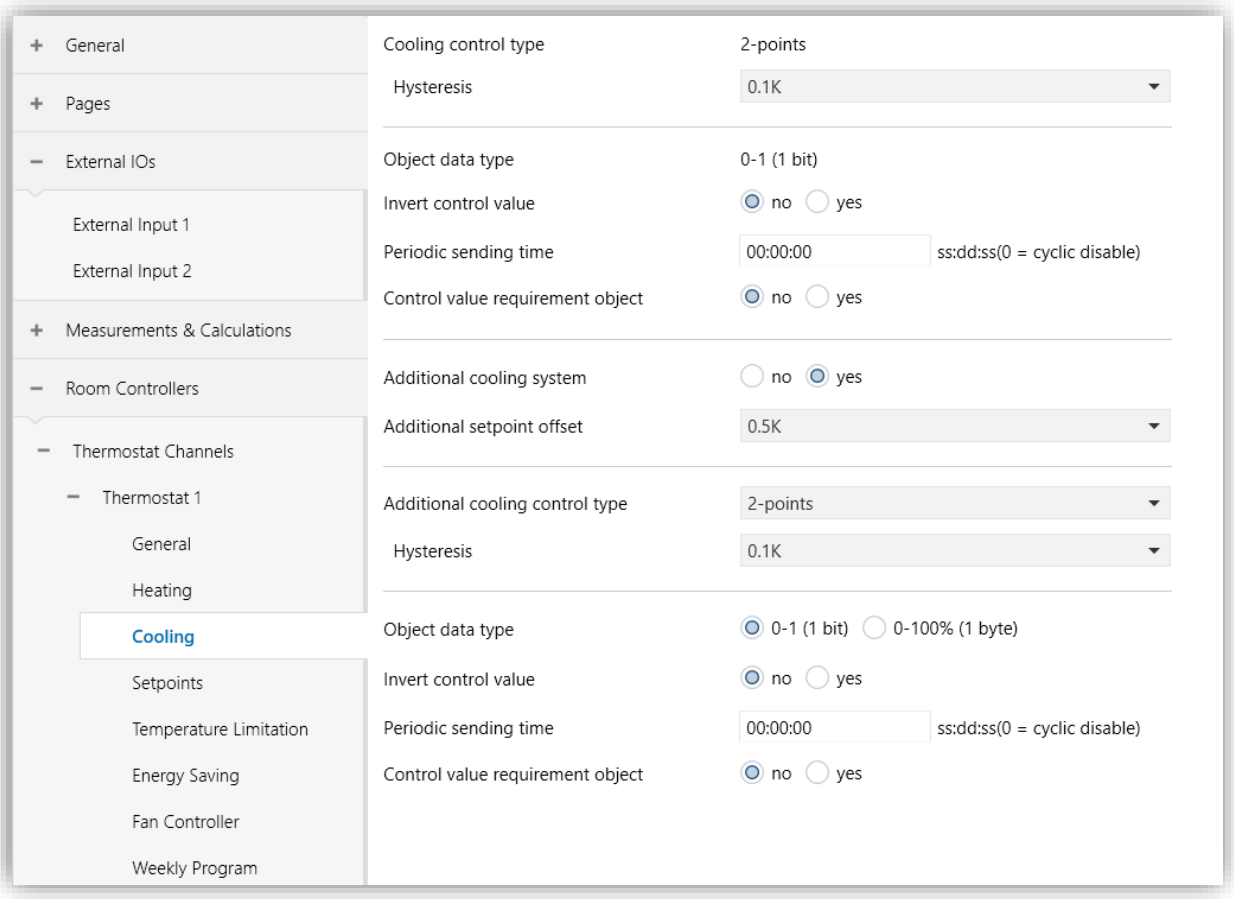


Fig. 51: Cooling 2-Points Control Configuration

4.7.5.2. Parameters List

PARAMETER	DESCRIPTION	VALUES
Cooling control type	This parameter determines the cooling control type.	**2 – points** PWM Continuous
Hysteresis	This parameter determines the hysteresis value.	**0.1K**…2.0K (°C) **0.18K**…3.6K (°F)
Invert control value	This parameter is used to invert control output.	**No** Yes
Periodic sending time	This parameter is used to periodically send the commands to the bus line.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the cooling system.	**No** Yes
Additional cooling system	This parameter activates the additional cooling system.	**No** Yes

4.7.5.3. Cooling PWM Control

The PWM (Pulse Width Modulation) proportional-integral controller allows the digital output to be set to On and Off by sampling an analogue control variable within a specified time. The controller runs periodically through a cycle and keeps its output ON for each period in proportion to the value of the control variable. By varying the ratio between the "ON" time and the "OFF" time of the heating system, the average activation time of the output changes, and as a result, the average heating power supplied by the room changes.

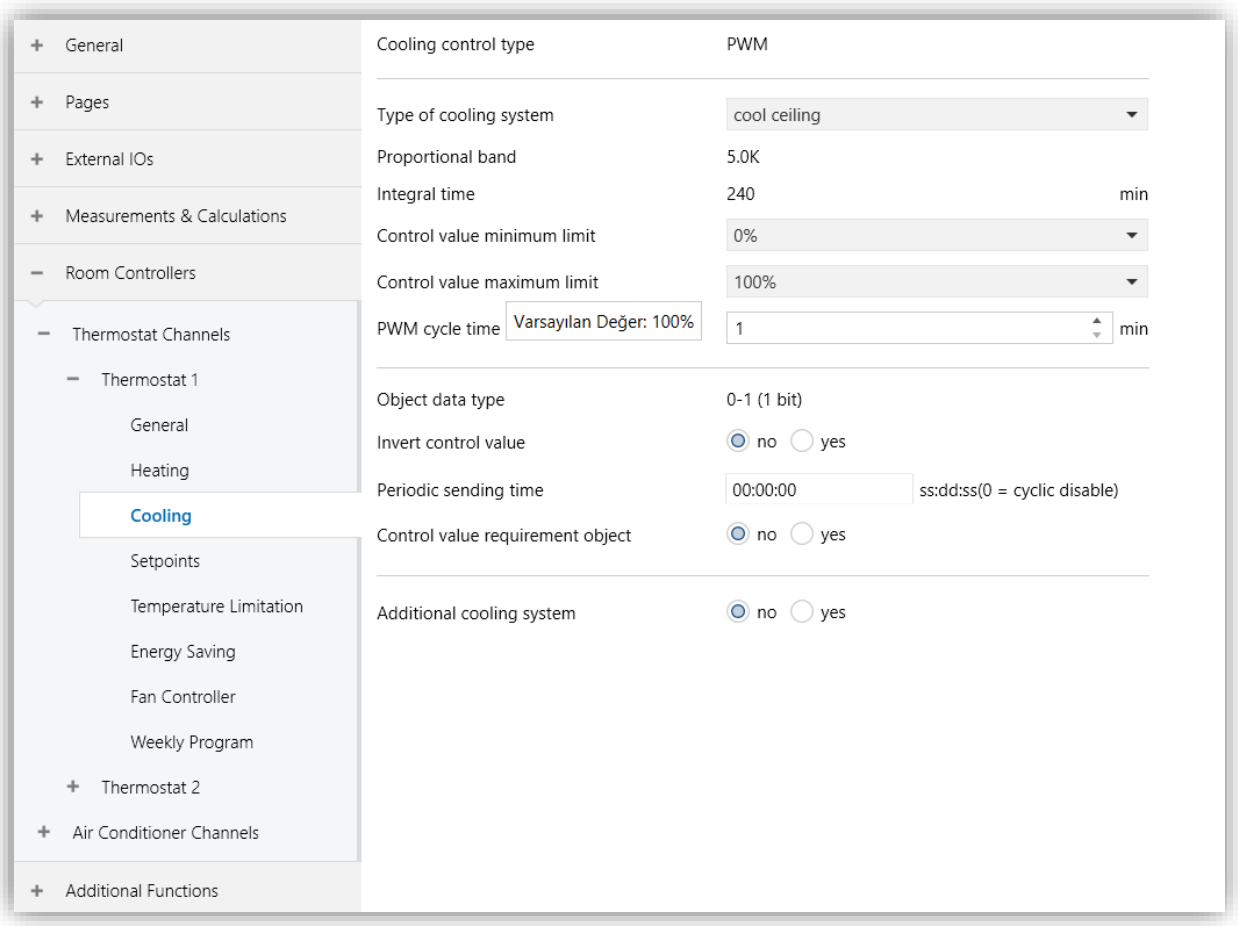


Fig. 52: Cooling PWM Control Configuration

4.7.5.4. Parameters List

PARAMETER	DESCRIPTION	VALUES
Type of cooling system	This parameter determines the cooling system to be controlled.	**Cool ceiling** Split unit Fan coil User defined
Proportional band (K)	This parameter determines the proportional band.	0.5K… **4.0K** … 10.0K (°C) 0.9K… **7.2K** … 18.0K (°F)
Integral time (min)	This parameter determines the integral time.	0…**90**…255
Control value minimum (%)	This parameter determines the output object's minimum control value.	**0%** (0%, 5%, 10%, 15%, 20%, 25%, 30%)
Control value maximum (%)	This parameter determines the output object's maximum control value.	**100%** (70%, 75%, 80%, 85%, 90%, %95, 100%)
PWM cycle time (min)	This parameter determines the PWM cycle time.	**1**…255
Object data type	This parameter is used to determine data type of control value object.	**0-1 (1 bit)** 0-100% (1 byte)
Invert control value	This parameter is used to invert control output.	**No** Yes
Periodic sending time	This parameter is used to periodically send the commands to the bus line.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the cooling system.	**No** Yes
Additional cooling system	This parameter activates the additional cooling system.	**No** Yes

4.7.5.5. Cooling Continuous Control

Proportional–integral control (PI control) is explained by the relationship shown below:

$$\boldsymbol{control\ variable(t)\ =\ Kp\ \times\ error(t)\ +\ Ki\ \times \int_0^t error(t)\ dt}$$

whereby:

$$error(t)\ =\ (Setpoint\ -Measured\ temperature)\ in\ heating$$
$$error(t)\ =\ (Measured\ temperature\ -Setpoint)\ in\ cooling$$
$$Kp\ =\ proportional\ constant$$
$$Ki\ =\ integral\ constant$$

The control variable contains integral and proportional (Ki and Kp) constants to eliminate errors. In practice, intuitively generated values are generally used.

Ex 1:

$$\boldsymbol{Proportional\ band\ BP\ [K] = \frac{100}{Kp}}$$

$$\boldsymbol{Integral\ time\ Ti\ [min] = \frac{Kp}{Ki}}$$

The proportional band is the error value that determines the maximum deflection output as 100%.

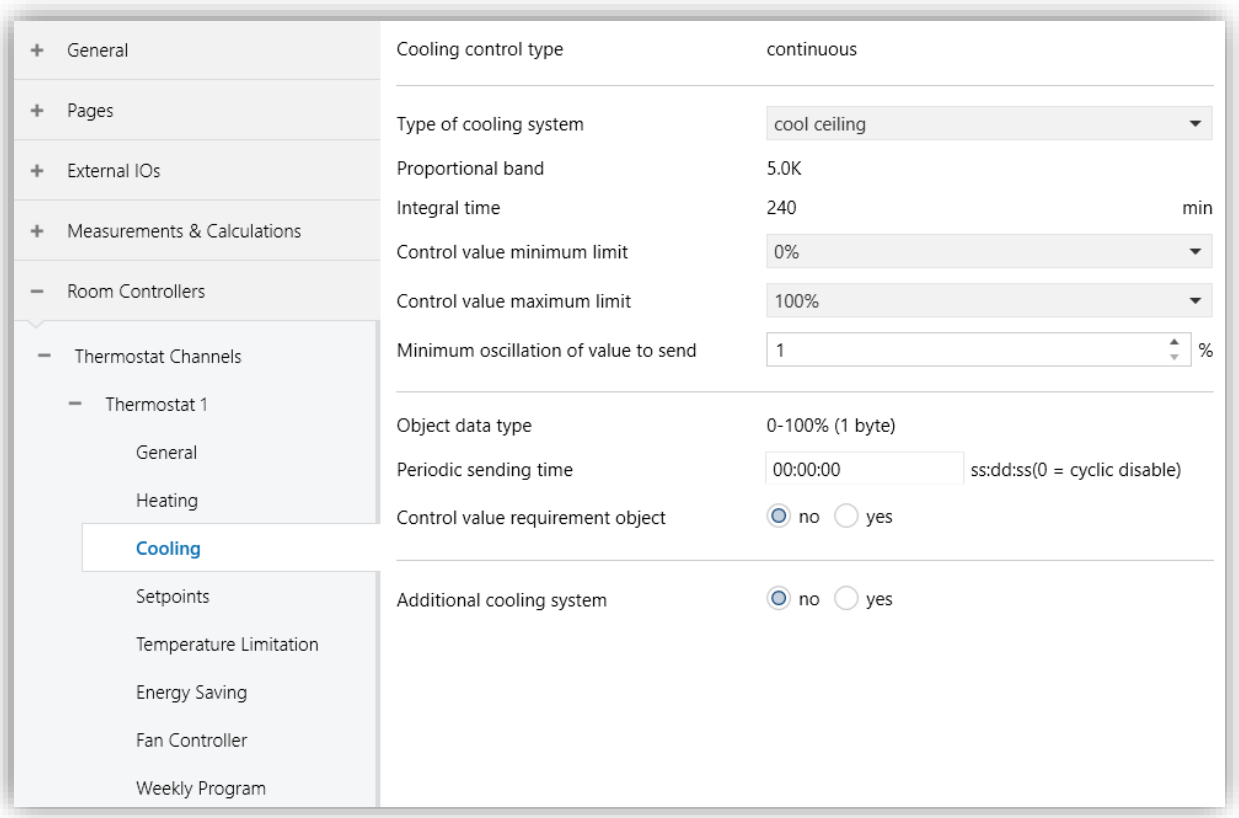


Fig. 53: Cooling Continuous Control Configuration

4.7.5.6. Parameters List

PARAMETER	DESCRIPTION	VALUES
Type of cooling system	This parameter determines the cooling system to be controlled.	**Cool ceiling** Split unit Fan coil User defined
Proportional band (K)	This parameter determines the proportional band.	0.5K… **5.0K** … 10.0K (°C) 0.9K… **9.0K** … 18.0K (°F)
Integral time (min)	This parameter determines the integral time.	0 … **90** … 255
Control value minimum (%)	This parameter determines the output object's minimum control value.	**0%** (0%, 5%, 10%, 15%, 20%, 25%, 30%)
Control value maximum (%)	This parameter determines the output object's maximum control value.	**100%** (70%, 75%, 80%, 85%, 90%, 95%, 100%)
Minimum oscillation of value to send (%)	This parameter determines the minimum oscillation value for the output object to send a value.	**1** …100
Periodic sending time	This parameter is used to periodically send the commands to the bus line.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the cooling system.	**No** Yes

4.7.5.7. Additional Cooling System

All types of cooling controls (2-points, PWM and continuous control) have additional cooling system options. The additional cooling system works in all control types with the same characteristics. The system activates itself according to the offset configuration. If ($T_{setpoint}$ + ΔT_{offset}) is higher than the ambient room temperature, the additional cooling system will be activated according to controller type.

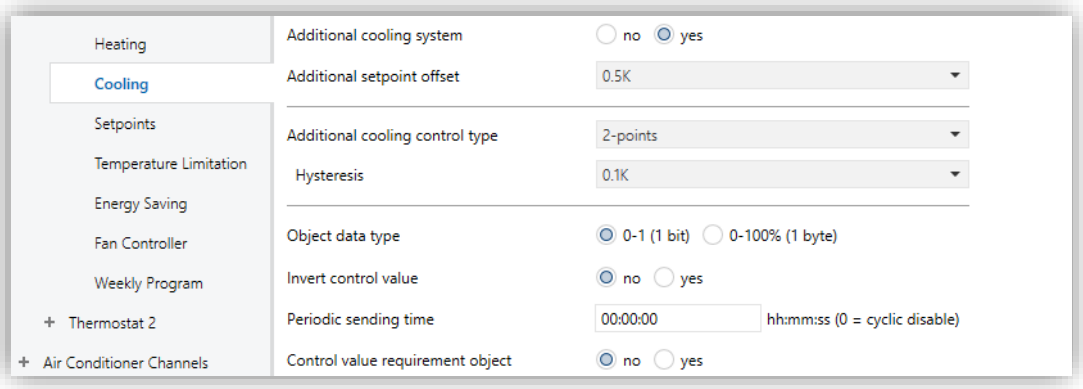


Fig. 54: Additional Cooling System Configuration

In additional cooling control, 2 - Points and PI Continuous controller cool the room until the difference between (T_{room} – $T_{setpoint}$) is equal to "Additional setpoint offset" parameter.

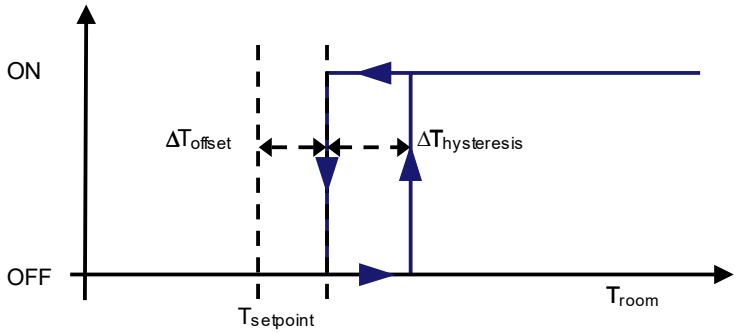


Fig. 55: 2 – Points Hysteresis Cycle for Additional Cooling Control

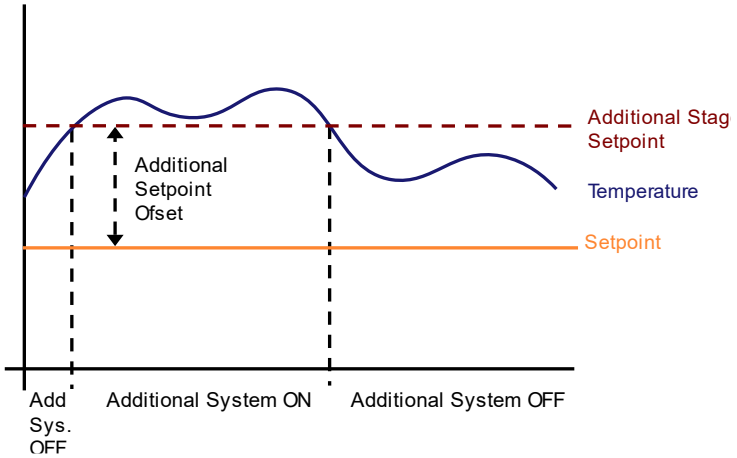


Fig. 56: PI Continuous Graph for Additional Cooling Control

4.7.5.8. Parameters List

PARAMETER	DESCRIPTION	VALUES
Additional setpoint offset	This parameter determines the difference between the setpoint temperature value and the additional cooling system's setpoint temperature value.	**0.5K** … 5.0K (°C) **0.9K** … 9.0K (°F)
Additional cooling control type	This parameter determines the additional cooling system's control object type.	**2 – points** PWM Continuous
Additional cooling control type: 2-points		
Hysteresis Value	This parameter determines the hysteresis value.	**0.1K**…2.0K (°C) **0.18K**…3.6K (°F)
Object type	This parameter determines the additional cooling system's object type.	**0-2 (1 bit)** 0-100% (1 byte)
Invert control value	This parameter is used to invert control output.	**No** Yes
Periodic sending time	This parameter determines the time of control value to be sent periodically.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the additional cooling system.	**No** Yes
Additional cooling control type: PWM		
Type of additional cooling system	This parameter determines the cooling system to be controlled.	**Cool ceiling** Split unit Fan coil User defined
Proportional band	This parameter determines the proportional band.	0.5K… **5.0K** … 10.0K (°C) 0.9K… **9.0K** … 18.0K (°F)
Integral time	This parameter determines the integral time.	0 … **240** … 255
Control value minimum limit	This parameter determines the output object's minimum control value.	**0%**, 5%, 10%, 15%, 20%, 25%, 30%

Control value maximum limit	This parameter determines the output object's maximum control value.	70%, 75%, 80%, 85%, 90%, %95, **100%**
PWM cycle time (min)	This parameter determines the PWM cycle time.	**1**…255
Object data type	This parameter is used to determine data type of control value object.	**0-2 (1 bit)** 0-100% (1 byte)
Invert control value	It is used to invert control output.	**No** Yes
Periodic sending time	This parameter is used to periodically send the commands to the bus line.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the additional cooling system.	**No** Yes
Additional cooling control type: Continuous		
Type of additional cooling system	This parameter determines the cooling system to be controlled.	Cool ceiling Split unit **Fan coil** User defined
Proportional band	This parameter determines the proportional band.	0.5K… **5.0K** … 10.0K (°C) 0.9K… **9.0K** … 18.0K (°F)
Integral time	This parameter determines the integral time.	0 … **240** … 255
Control value minimum limit	This parameter determines the output object's minimum control value.	**0%**, 5%, 10%, 15%, 20%, 25%, 30%
Control value maximum limit	This parameter determines the output object's maximum control value.	70%, 75%, 80%, 85%, 90%, %95, **100%**
Minimum oscillation of value to send	This parameter determines the minimum oscillation value for the output object to send a value.	**1** … 100
Periodic sending time	This parameter is used to periodically send the commands to the bus line.	**00:00:00** … 18:12:15
Control value requirement object	This parameter is used to send status information about the controller value of the additional cooling system.	**No** Yes

4.7.6. Thermostat - Heating & Cooling

Heating & Cooling mode is generally used when there are 2 different heating and cooling sources or only 1 source that has both heating and cooling ability together. If the heating/cooling sources are different, the command value object parameter should be selected as "2 separated objects". However, if heating and cooling are obtained from the same source, the command value object parameter should be selected as "1 common object". Additionally, in this mode, the distinction is made whether the switch-over between heating and cooling is to be affected automatically or in a controlled way through the communication object.

In the automatic switch-over option: for the heating, the controller will turn on the heating when the room temperature has fallen below a preset dead band limit. As soon as the room temperature is exceeding the heating setpoint, the control will turn off the heating in the heating & cooling mode. For the cooling, the controller will turn on the cooling system when the room temperature has exceeded a preset dead band limit. As soon as the room temperature is reaching above the cooling setpoint, the control will turn off the cooling system in the heating & cooling mode.

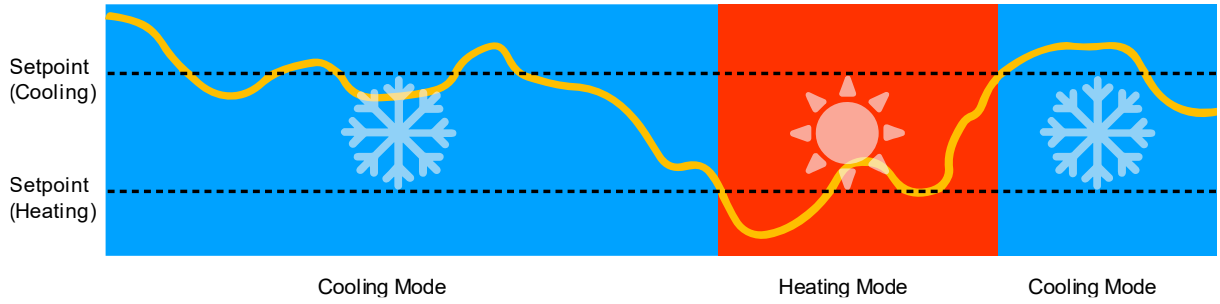


Fig. 57: Automatic Heating & Cooling Mode Switch

For a proper behaviour of the automatic switch function, the setpoint of the Cooling mode is required to be higher than that of the Heating mode.

In via communication object option: In this option, there is no dead band concept compared to the automatic option. The main difference between automatic and communication object options; the mode switch-over between modes is made manually.

4.7.6.1. Parameters List

In heating & cooling mode, cooling configurations and heating configurations can be made separately mentioned before. In this section, only extra parameters for this mode are described below.

PARAMETER	DESCRIPTION	VALUES
Thermostat mode	The thermostat mode's operating type is determined with this parameter.	**Master** Slave
Temperature source	This parameter determines whether the temperature source is external or internal.	**Internal temperature** Temperature object Calculation 1…6
Room controller mode	Room controller mode is determined with this parameter.	**Heating** Cooling Heating & Cooling
Command value object	The object types of temperature command values for heating and cooling mode are determined with this parameter.	**Common** Separated
Switch-over heating/cooling	This parameter determines how the heating/cooling transition is made.	Via object **Automatic**
Room controller mode after reset	This parameter determines the room controller mode after the device restarts.	**Previous mode** Heating Cooling

4.7.7. Thermostat - Set Points

Temperature setpoints for heating or cooling modes are configured in this section. The operation modes such as comfort, standby, night and frost protection of "heating", "cooling" and "heating & cooling" modes can be separately specified from this section. The temperature setpoint value can be configured to send to the KNX bus line with 4 different settings such as "Disable", "Periodically", "On change" and "Periodically and on change". Besides, how much the maximum bandwidth setting will be configured for that increasing or decreasing the temperature value manually can be determined. Moreover, it is possible to set which setpoint values will be used when there is a power failure.

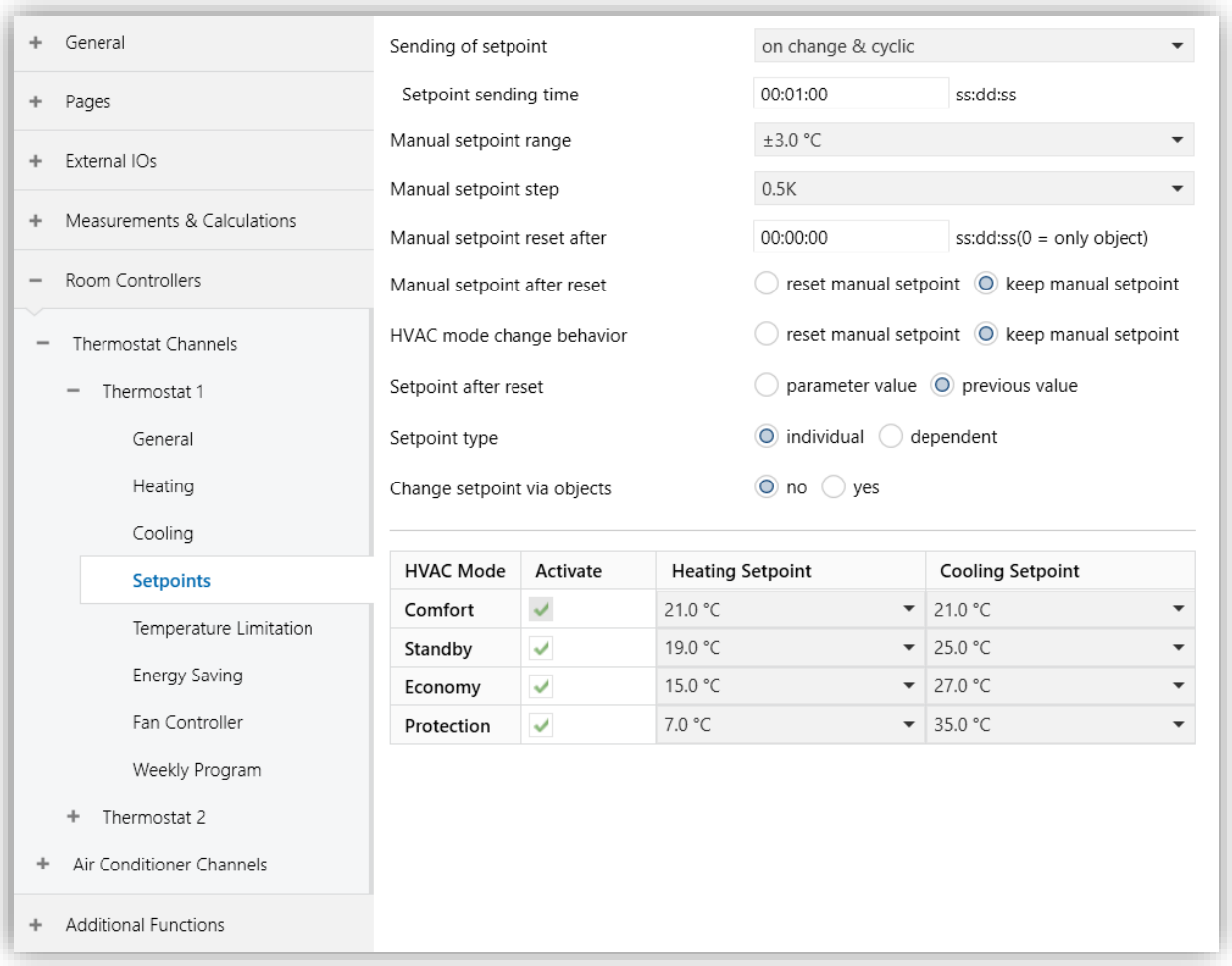


Fig. 58: Set Points Configuration

Note: If Heating/Cooling automatic mode is used HVAC mode setpoints must be in the range of manual setpoint. Otherwise, shifts in setpoints may occur in automatic heating-cooling transitions.

Note: Heating and Cooling setpoints limited with 10°C to 40°C for Comfort, Standby and Economy modes, 0°C to 15.5°C for frost protection mode and 25°C to 45°C for heat protection mode. User can change setpoint temperature bases with this ranges. If an attempt is made to apply a setpoint base other than the limits from the setpoint base objects, the limit value becomes valid.

4.7.7.1. Parameters List

PARAMETER	DESCRIPTION	VALUES
Sending of setpoint	This parameter allows sending the setpoint temperature value information. **On change:** The Temperature value information is sent when the setpoint temperature value changes by 1 K. **Periodically:** The Temperature value information is sent periodically. **Periodically and on change:** The Temperature value information is sent periodically or when the setpoint temperature value changed 1 K.	Disable **On change** Cyclic On change & cyclic
Setpoint sending time[1]	This parameter determines the time of the setpoint temperature value to be sent periodically.	
Manual setpoint range	This parameter configures the maximum and minimum limit values for the setpoint temperature value.	±1.0 ... **±3.0** ... ±10.0 (˚C) ±1.8 ... **±5.4** ... ±22.5 (˚F)
Manual setpoint step	This parameter configures the maximum and minimum limit values for the setpoint temperature value.	0.1K … **0.5K** … 3.5K (˚C) 0.18K … **0.9K** … 6.3K (˚F)
Manual setpoint reset after	This parameter determines the time of value to be sent setpoint reset after.	**00:00:00** … 18:12:15
Manual setpoint after reset	This parameter determines the behaviour of the manual setpoint's value after device reset. **Reset manual setpoint:** The manual setpoint is reset after device reset. **Keep manual setpoint:** The manual setpoint is continued after device reset.	Reset manual setpoint **Keep manual setpoint**
HVAC mode change behaviour	This parameter determines the behaviour of the manual setpoint's value after receiving the new set mode. **Reset manual setpoint:** The manual setpoint is reset after the new setting mode is received with this option.	Reset manual setpoint **Keep manual setpoint**

	Keep manual setpoint: The manual setpoint is continued after the new setting mode is received with this option.	
Setpoint after reset	This parameter determines the setpoint temperature after a reset for any reason, such as power failure.	Parameter value **Previous value**
Setpoint type	The desired temperature value can be controlled with individual or dependent setpoints by this parameter. If dependent mode is selected the setpoints of comfort and protect can be configured as individual setpoint. Standby and economy mode's setpoints can be configured as dependent setpoint. Even dependent mode is selected, all of the operation mode's setpoints can be change via object separately. So, if the comfort's setpoint is changed economy or standby's setpoints aren't updated according to comfort setpoint.	**Individual** Dependent
Change setpoint via objects	With this parameter, setpoint objects for all operation mode are visible.	**No** Yes
Comfort Mode Activate	This parameter is used to determine the activation of comfort mode. If this parameter is checked, comfort mode can be useable.	**Checked** Unchecked
Comfort Mode Heating Setpoint (˚C)	The desired temperature value for comfort mode is configured with this parameter.	10.0 … **21.0** … 40 (˚C) 50.0 … **69.8** … 104 (˚F)
Comfort Mode Cooling Setpoint (˚C)	The desired temperature value for comfort mode is configured with this parameter.	10.0 … **21.0** … 40 (˚C) 50.0 … **69.8** … 104 (˚F)
Standby Mode Activate	This parameter is used to determine the activation of standby mode. If this parameter is checked, standby mode can be useable.	**Checked** Unchecked
Standby Mode Heating Setpoint (˚C)	The desired temperature value of heating for standby mode is configured with this parameter.	10.0 … **19.0** … 40 (˚C) 50.0 … **66.2** … 104 (˚F)
Standby Mode Cooling Setpoint (˚C)	The desired temperature value for standby mode is configured with this parameter.	10.0 … **25.0** … 40 (˚C) 50.0 … **77.0** … 104 (˚F)

Economy Mode Activate	This parameter is used to determine the activation of economy mode. If this parameter is checked, economy mode can be useable.	**Checked** Unchecked
Economy Mode Heating Setpoint (˚C)	The desired temperature value of heating for economy mode is configured with this parameter.	10.0 … **15.0** … 40 (˚C) 50.0 … **59.0** … 104 (˚F)
Economy Mode Cooling Setpoint (˚C)	The desired temperature value of cooling for economy mode is configured with this parameter	10.0 … **27.0** … 40 (˚C) 50.0 … **80.6** … 104 (˚F)
Protection Mode Activate	This parameter is used to determine the activation of protection mode. If this parameter is checked, protection mode can be useable.	**Checked** Unchecked
Protection Mode Heating Setpoint (˚C)	The desired temperature value of heating for protection mode is configured with this parameter.	0.0 … **7.0** … 15.5 (˚C) 32.0…**44.6** … 59.9 (˚F)
Protection Mode Heating Setpoint (˚C)	The desired temperature value of cooling for protection mode is configured with this parameter	25.0…**35.0**…45.0 (˚C) 77.0…**95.0**…113.0 (˚F)

[1] This parameter is visible when the parameter "Sending of setpoint" is set to "Periodically" or "periodically and on change".

4.7.8. Thermostat – Temperature Limitation

Using the limit temperature, the controller's control value for this stage can be set to 0 on reaching a parameterized temperature. In this way, exceeding (heating) or dropping below (cooling) this temperature can be prevented. An example of the usage of the limit temperature is floor heating, where exceeding a specific temperature must be prevented to protect the material of the floor.

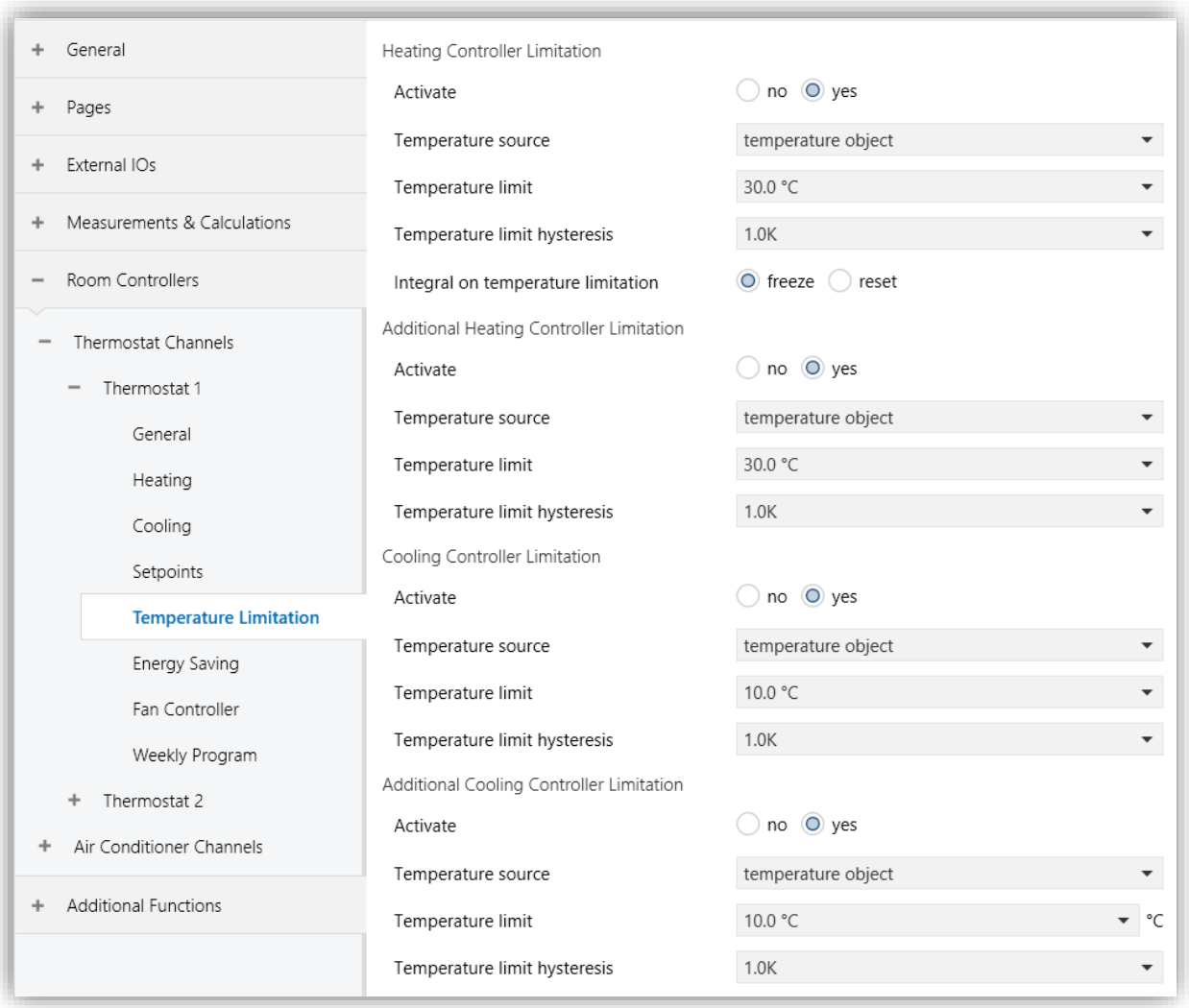


Fig. 59: Temperature Limitation Configuration

4.7.8.1. Parameters List

PARAMETER	DESCRIPTION	VALUES
Heating Controller Limitation Activate	This parameter is used to activate limit temperature for heating controller.	**No** Yes
Heating Controller Limitation Activate: Yes		
Temperature Source	This parameter is used to determine the source of temperature for limitation function. It is not suitable to use the same temperature sensor for the measurement of the room temperature and for the measurement of the limit temperature.	Internal temperature **Temperature object** Calculation 1…6
Temperature Limit	This parameter is used to determine the limit temperature that is not allowed to be exceeded (heating). If the temperature reaches this value, the control value is immediately set to 0.	1… **30** …60 (°C) 32… **86** …140 (°F)
Temperature Limit Hysteresis	This parameter is used to determine the hysteresis on the limit temperature specifies the value by which the limit temperature must be dropped below again (heating) before the controller becomes active again.	0.5K … **1K** … 5K (°C) 0.9K … **1.8K** … 9K (°F)
Integral on temperature limitation[1]	This parameter is used to decide what is to happen to the I-proportion on reaching the limit temperature. **Freeze:** Keeps the current accumulated error caused by I-proportion. **Reset:** Resets the accumulated error caused by I-proportion.	**Freeze** Reset
Additional Heating Controller Limitation Activate	This parameter is used to activate limit temperature for additional heating controller.	**No** Yes
Additional Heating Controller Limitation Activate: Yes		
Temperature Source	This parameter is used to determine the source of temperature for limitation function. It is not suitable to use the same temperature sensor for the measurement of the room temperature and for the measurement of the limit temperature.	Internal temperature **Temperature object** Calculation 1…6
Temperature Limit	This parameter is used to determine the hysteresis on the limit temperature specifies the value by which the limit temperature must be dropped below again	1… **30** …60 (°C) 32… **86** …140 (°F)

	(heating) before the controller becomes active again.	
Temperature Limit Hysteresis	This parameter is used to determine the hysteresis on the limit temperature specifies the value by which the limit temperature must be dropped below again (heating) before the controller becomes active again.	0.5K … **1K** … 5K (°C) 0.9K … **1.8K** … 9K (°F)
Integral on temperature limitation[2]	This parameter is used to decide what is to happen to the I-proportion on reaching the limit temperature. **Freeze:** Keeps the current accumulated error caused by I-proportion. **Reset:** Resets the accumulated error caused by I-proportion.	**Freeze** Reset
Cooling Controller Limitation Activate	This parameter is used to activate limit temperature for cooling controller.	**No** Yes
Cooling Controller Limitation Activate: Yes		
Temperature Source	This parameter is used to determine the source of temperature for limitation function. It is not suitable to use the same temperature sensor for the measurement of the room temperature and for the measurement of the limit temperature.	Internal temperature **Temperature object** Calculation 1…6
Temperature Limit	This parameter is used to determine the limit temperature that is not allowed to be dropped below (cooling). If the temperature reaches this value, the control value is immediately set to 0.	1… **10** …60 (°C) 32… **50** …140 (°F)
Temperature Limit Hysteresis	This parameter is used to determine the hysteresis on the limit temperature specifies the value by which the limit temperature must be exceeded (cooling) before the controller becomes active again.	0.5K … **1K** … 5K (°C) 0.9K … **1.8K** … 9K (°F)
Integral on temperature limitation[3]	This parameter is used to decide what is to happen to the I-proportion on reaching the limit temperature. **Freeze:** Keeps the current accumulated error caused by I-proportion. **Reset:** Resets the accumulated error caused by I-proportion.	**Freeze** Reset
Additional Cooling Controller Limitation Activate	This parameter is used to activate limit temperature for additional cooling controller.	**No** Yes

Additional Cooling Controller Limitation Activate: Yes		
Temperature Source	This parameter is used to determine the source of temperature for limitation function. It is not suitable to use the same temperature sensor for the measurement of the room temperature and for the measurement of the limit temperature.	Internal temperature **Temperature object** Calculation 1…6
Temperature Limit	This parameter is used to determine the limit temperature that is not allowed to be dropped below (cooling). If the temperature reaches this value, the control value is immediately set to 0.	1… **10** …60 (°C) 32… **50** …140 (°F)
Temperature Limit Hysteresis	This parameter is used to determine the hysteresis on the limit temperature specifies the value by which the limit temperature must be exceeded (cooling) before the controller becomes active again.	0.5K … **1K** … 5K (°C) 0.9K … **1.8K** … 9K (°F)
Integral on temperature limitation[4]	This parameter is used to decide what is to happen to the I-proportion on reaching the limit temperature. **Freeze:** Keeps the current accumulated error caused by I-proportion. **Reset:** Resets the accumulated error caused by I-proportion.	**Freeze** Reset

[1] This parameter is visible when heating controller type is set to "PWM" or "Continuous".
[2] This parameter is visible when additional heating controller type is set to "PWM" or "Continuous".
[3] This parameter is visible when cooling controller type is set to "PWM" or "Continuous".
[4] This parameter is visible when additional cooling controller type is set to "PWM" or "Continuous".

4.7.9. Thermostat – Energy Saving

In order to realize energy-saving functions, **window contacts** (to detect the opening of windows or doors), **presence and movement sensors** and **card holders** can be used.

The Energy saving folder includes:

- Window contacts
- Presence sensors
- Card holder

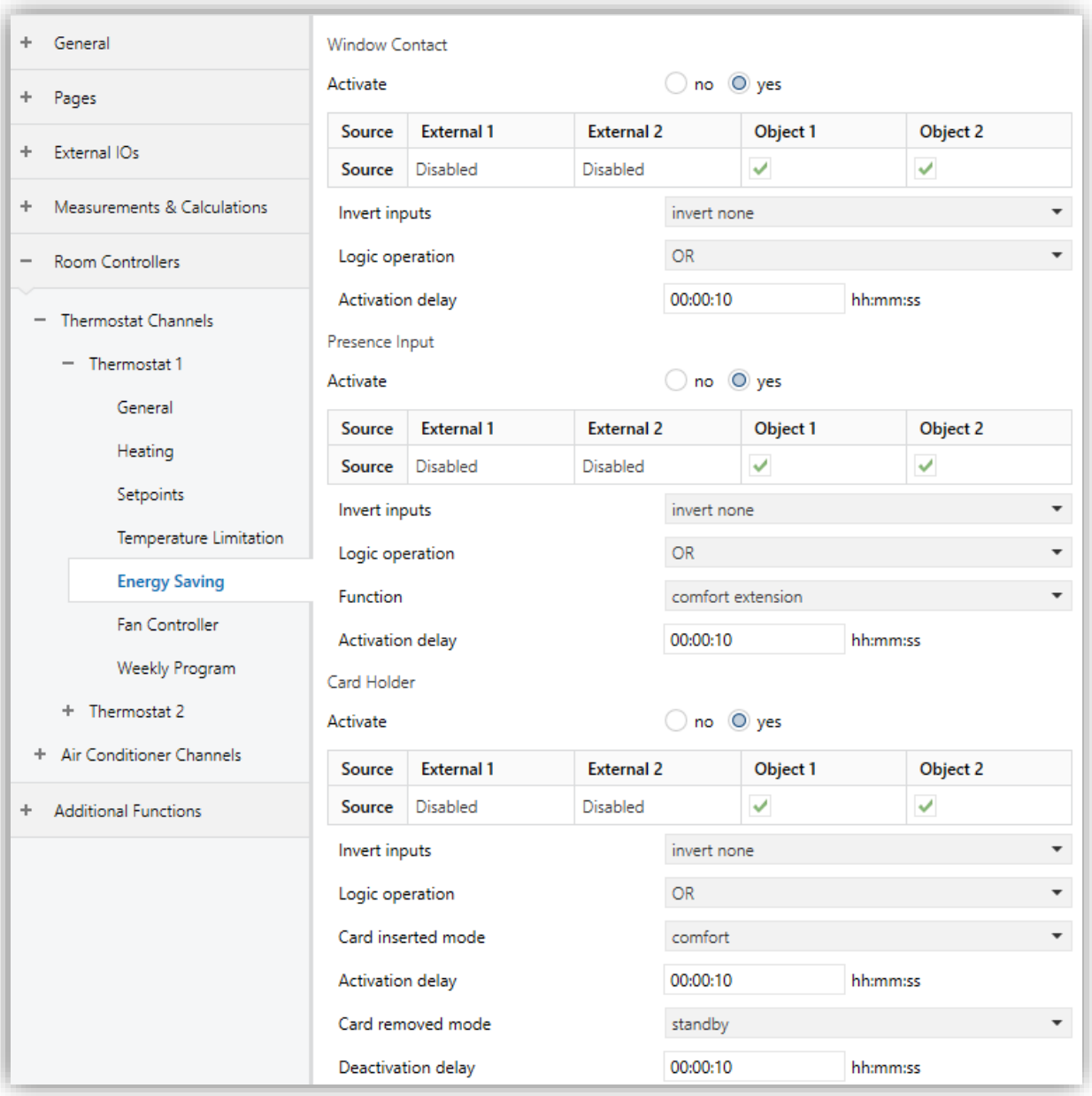


Fig. 60: Energy Saving Configuration

4.7.9.1. Window Contacts

In order to realize energy-saving functions, window contacts (to detect the opening of windows or doors) can be used. The device can acquire the status of a contact by means of an external(digital) input or receive the status of two objects connected to different KNX devices (binary inputs, pushbutton interfaces). When a window opens, the device automatically switches to Protection operating mode; when it closes, the device automatically returns to the previous operating mode. When acquiring more than one signal, they can be combined in logical OR, AND or XOR according to the "Logic operation" parameter.

When an open window is detected, the operating mode is forced into building protection and remains forced until all windows are closed. The application program features an activation time parameter for opening delay to discriminate between an occasional, short opening and a long opening, which justifies the energy saving mode recall.

After activation delay the operation mode is changed. Also, if fan control is enabled for the current thermostat control, the fan mode is set "Auto". The operation mode, fan level and fan mode cannot be updated via KNX bus. Window contact function has the highest priority between thermostat energy saving functions. Also, has absolute priority over the operating mode forced by the weekly program and over the HVAC mode forced by supervisor through the communication object HVAC Forced mode in DPT 20.102

If the operation mode input object is received a telegram, operation mode's icon flashes on the LCD for the end-users to indicate "energy saving" activated. And if the fan level or fan mode input object is received a telegram, locked icon flashes on the LCD for the end-users to indicate that this function is locked because of energy saving. These cases continue until the window contact input object is received a telegram or the external input is trigged for deactivation the energy saving.

4.7.9.2. Parameters List

PARAMETER	DESCRIPTION	VALUES
Window Contact Activate	This parameter is used to enable window contact function for thermostat energy saving. Two 1-bit objects are visible when it is enabled.	**No** Yes
Window Contact Activate: Yes		
External X Source	If this parameter is checked, the External X input is selected as a trigger for the window contact function to be activated.	**Disabled** Checked Unchecked
Object X Source	If this parameter is checked, the Object X is selected as a trigger for the window contact function to be activated.	**Checked** Unchecked
-> Invert inputs[1]	This parameter determines how to use the input objects.	**Invert none** Invert input 1 / 2 Invert both
Logic operation	This parameter is used to combine the inputs with logical operations.	**OR** AND XOR
Activation delay	This parameter is used to determine activation delay time. The energy saving function is delayed for the specified time.	00:00:00… **00:00:10** …18.12.15

[1] This parameter is visible when the parameter "Object X Source" is set to "Checked".

4.7.9.3. Presence Input

Presence input function includes a set of optional features, oriented to energy saving, which become available when the device is configured as integrated controller. This function is **only active if the actual operating mode is set to comfort.**

Generally speaking, if a human presence is detected and limited to the occupancy period, the comfort operating mode can be extended; vice versa, if no presence is detected, the comfort operating mode can be limited, because no longer necessary.

If one of the presence inputs objects or the external input selected as presence input, is received a telegram, it is triggered for activation. When acquiring more than one signal, they can be combined in logical OR, AND or XOR according to the "Logic operation" parameter.

In case a forced HVAC mode is used by supervisor through the communication object HVAC forced mode in DPT 20.102, the forced operating mode has a higher priority compared to the mode foreseen by the presence input function, so it will prevail.

In case the energy saving function is carried out through window contacts, the system switches to building protection mode when detecting an open window. Window contact function has a higher priority compared to both the forced mode and the mode foreseen by the presence input function.
There are three presence input function modes: **comfort extension**, **comfort limitation** and a combination of these two modes called **comfort extension and limitation**.

Comfort extension:
If during this time, a presence is detected, the operating mode remains comfort except for even if the operating mode is forced by the user or the weekly program function shifts to economy or standby. However, If the operation mode is set to protection, energy saving mode is interrupted until when the operation mode is comfort again while the presence input is still active. In this case, or if the operation mode input object is received a telegram except for "comfort" and "protection", the telegram is saved to use after comfort extension.
If the operation mode input object is received a telegram, the operation mode's icon flashes on the LCD for the end-users to indicate "energy saving" activated. If the fan level or fan mode input object is received a telegram, the locked icon flashes on the LCD for the end-users to indicate that this function is locked because of energy saving.

Comfort limitation:
If one of the presence input objects is set or the digital input selected as presence input is triggered for deactivation (the end-users left the room) and the operation mode is not forced, after the function active time, the operation mode is set from "Comfort" to the mode which is determined via "Limitation mode" parameter until the presence input object is set or the digital input selected as presence input is trigged for activation (the end-users entered the room). If the operation mode is set to protection, energy saving mode is interrupted until when the operation mode is comfort again while the presence input is still inactive.

Comfort extension and comfort limitation:
This mode is a combination of comfort extension and comfort limitation modes.

4.7.9.4 Parameters List

PARAMETER	DESCRIPTION	VALUES
Presence Input Activate	This parameter is used to enable presence input function for thermostat energy saving. Two 1-bit objects are visible when it is enabled.	**No** Yes
Presence Input Activate: Yes		
External X Source	If this parameter is checked, the External X input is selected as a trigger for the presence input function to be activated.	**Disabled** Checked Unchecked
Object X Source	If this parameter is checked, the Object X is selected as a trigger for the presence input function to be activated.	**Checked** Unchecked
-> Invert inputs[1]	This parameter determines how to use the input objects.	**Invert none** Invert input 1 / 2 Invert both
Logic operation	This parameter is used to combine the inputs with logical operations.	**OR** AND XOR
Activation delay	This parameter is used to determine activation delay time. The energy saving function is delayed for the specified time.	00:00:00… **00:00:10** …18.12.15
Function	This parameter is used to determine how to use the energy saving function for presence input.	**Comfort extension** Comfort limitation Comfort extension and limitation
-> Limitation mode[2]	It is used to determine the operation modes which will be set for the energy saving function in comfort limitation mode.	**Comfort – standby** Comfort - economy

[1] This parameter is visible when the parameter "Object X Source" is set to "Checked".

[2] This parameter is visible when the parameter "Function" is set to "Comfort limitation" or "Comfort extension and limitation".

4.7.9.5. Card Holder

If the card holder input object is set or the digital input selected as card holder input is triggered for activation (the end-users entered the room with card) and comfort extension mode is not active and the operation mode is not forced, then the operation mode is set as the mode via "Card insertion HVAC mode" parameter after the function active time via "Activation delay on card insertion" parameter.

Otherwise, if card holder input object is set or the digital input selected as card holder input is triggered for deactivation (the end-users left the room with card) and comfort extension mode is not active and the operation mode is not forced, then the operation mode is set as the mode via "Card removal HVAC mode" parameter after the function active time via "Activation delay on card removal" parameter.

When acquiring more than one signal, they can be combined in logical OR, AND or XOR according to the "Logic operation" parameter.

Card holder function has the lowest priority between thermostat energy saving functions.

4.7.9.6 Parameters List

PARAMETER	DESCRIPTION	VALUES
Card Holder Activate	This parameter is used to enable card holder function for thermostat energy saving. Two 1-bit objects are visible when it is enabled.	**No** Yes
Card Holder Activate: Yes		
External X Source	If this parameter is checked, the External X input is selected as a trigger for card holder function to be activated.	**Disabled** Checked Unchecked
Object X Source	If this parameter is checked, the Object X is selected as a trigger for the card holder function to be activated.	**Checked** Unchecked
-> Invert inputs[1]	This parameter determines how to use the input objects.	**Invert none** Invert input 1 / 2 Invert both
Logic operation	This parameter is used to combine the inputs with logical operations.	**OR** AND XOR
Card inserted mode	This parameter is used to determine the operation mode which is set, when card insertion.	Auto **Comfort** Standby Economy Protection
Activation delay	This parameter is used to determine the activation delay time for card insertion.	00:00:00… **00:00:10** …18.12.15
Card removed mode	This parameter is used to determine the operation mode which is set, when card removal.	Auto Comfort **Standby** Economy Protection
Deactivation delay	This parameter is used to determine the activation delay time for card removal.	00:00:00… **00:00:10** …18.12.15

[1] This parameter is visible when the parameter "Object X Source" is set to "Checked".

4.7.10. Thermostat – Fan Controller

If the parameter "Fan control used for room control" is set to "Enabled" from the "General" parameter page, the configuration page that is related to fan controller is now opened as "Fan Controller" under the "Room Controller" parameter page instead of the "LCD" parameter page.

The configuration settings in this section are configured such as, the selection of the fan speed level of the device to be used, the fan speed transitions in regard to the percentage value to be changed, the fan controller type selection, delay time for starting and delay time for stopping the fan and other arrangements related to fan control.

4.7.10.1. Fan 2-Points Control

This type of fan control is similar to the 2 points control with hysteresis: the fan speed is activated/deactivated according to the difference between the desired temperature and the measured temperature. The relevant difference with the 2 points algorithms with hysteresis is that, in this case, there is not a single stage on which the hysteresis loop is executed, by setting the thresholds for switching on and off of the speed, but five stages may exist.

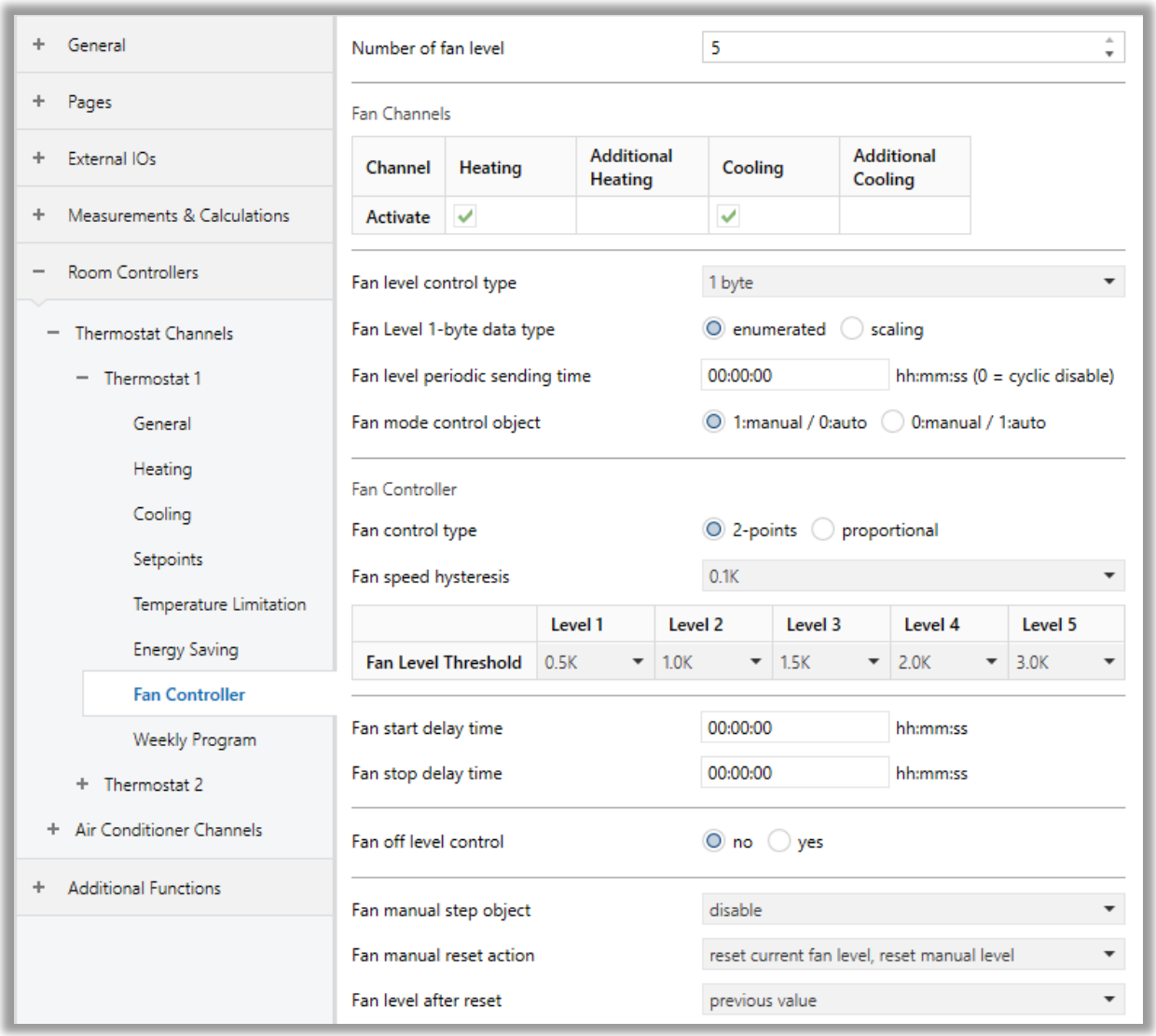


Fig. 61: Fan Controller 2-Points Control Configuration

This means that a speed level corresponds to each stage and when the difference between the measured temperature and the desired temperature causes the activation of a further speed.

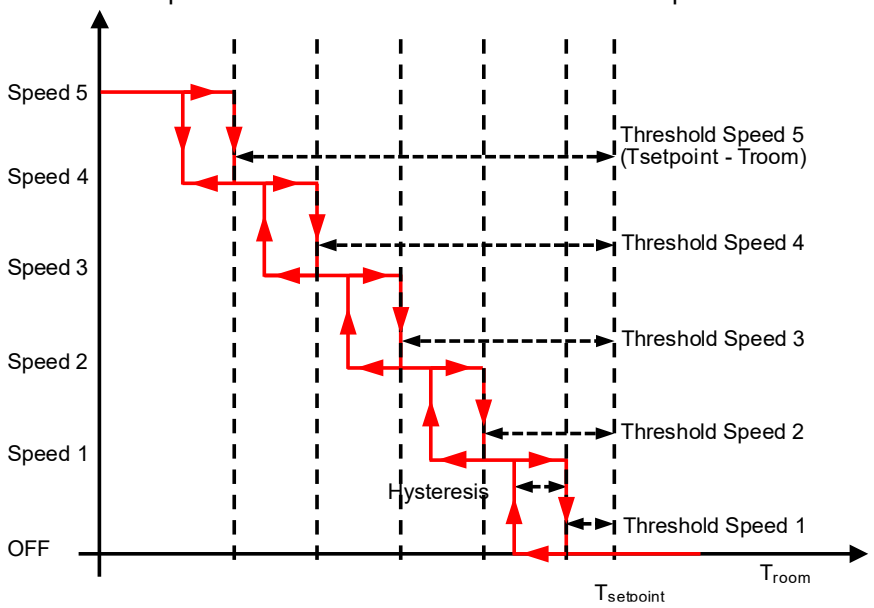


Fig. 62: Fan Controller 2-Points Control Cycle for Heating

The figure in the above graph refers to the speed control of the fan with three operating stages as regards the heating. Looking at the graph, it has to be noted that for each stage there is a hysteresis loop, as well as at any speed are assigned two thresholds which determine the activation and deactivation. The thresholds are determined by the values set in the application program and can be summarized as follows:

- Speed 1 (1st stage) – The speed is turned ON when the value of the room temperature is lower than the value (T_{set} – Threshold Speed1 – hysteresis) and turned OFF when the room temperature value reaches the value (T_{set} – Threshold Speed1); the first speed is also switched OFF when a higher speed must be turned ON. The default value for the parameter Threshold Speed1 = 0 K.
- Speed 2 (2nd stage) – The speed is turned ON when the value of the room temperature is lower than the value (T_{set} – Threshold Speed2 – hysteresis) and turned OFF when the room temperature value reaches the value (T_{set} – Threshold Speed 2); the second speed is also switched OFF when Speed 3 must be turned ON.
- Speed 3 (3rd stage) – The speed is turned ON when the value of the room temperature is lower than the value (T_{set} – Threshold Speed3 – hysteresis) and turned OFF when the room temperature value reaches the value (T_{set} – Threshold Speed 3).
- Speed 4 (4rd stage) – The speed is turned ON when the value of the room temperature is lower than the value (T_{set} – Threshold Speed 4 – hysteresis) and turned OFF when the room temperature value reaches the value (T_{set} – Threshold Speed 4).
- Speed 5 (5rd stage) – The speed is turned ON when the value of the room temperature is lower than the value (T_{set} – Threshold Speed 5 – hysteresis) and turned OFF when the room temperature value reaches the value (T_{set} – Threshold Speed 5).

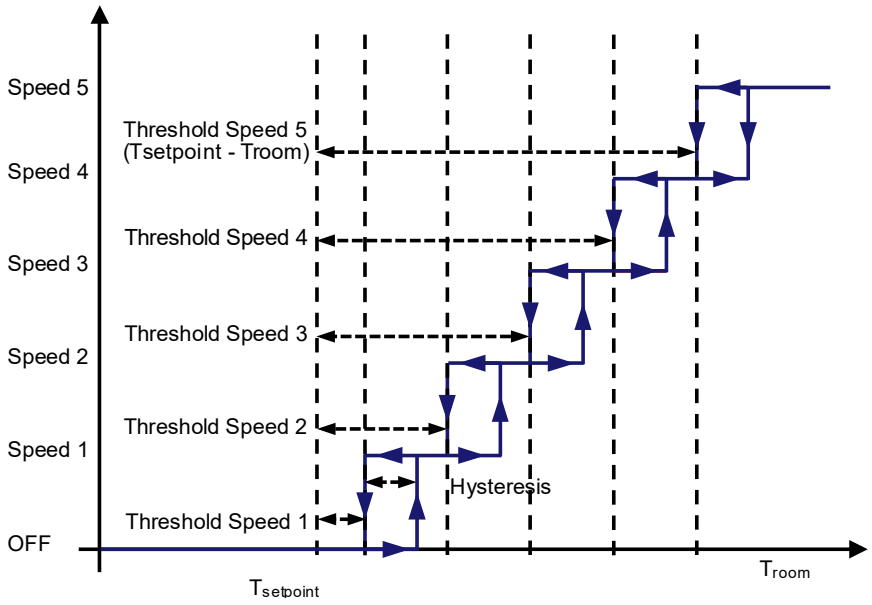


Fig. 63: Fan Controller 2-Points Control Cycle for Cooling

The figure in the above graph refers to the speed control of the fan with three operating stages as regards the cooling. Looking at the graph, it has to be noted that for each stage there is a hysteresis loop, as well as at any speed are assigned two thresholds which determine the activation and deactivation. The thresholds are determined by the values set in the application program and can be summarized as follows:

- Speed 1 (1st stage) – The speed is turned ON when the value of the room temperature is higher than the value (T_{set} + Threshold Speed1 + hysteresis) and turned OFF when the room temperature value reaches the value (T_{set} + Threshold Speed1); the first speed is also switched OFF when a higher speed must be turned ON. The default value for the parameter Threshold Speed1 = 0 K.
- Speed 2 (2nd stage) – The speed is turned ON when the value of the room temperature is higher than the value (T_{set} + Threshold Speed2 + hysteresis) and turned OFF when the room temperature value reaches the value (T_{set} + Threshold Speed2); the second speed is also switched OFF when Speed 3 must be turned ON.
- Speed 3 (3rd stage) – The speed is turned ON when the value of the room temperature is higher than the value (T_{set} + Threshold Speed3 + hysteresis) and turned OFF when the room temperature value reaches the value (T_{set} + Threshold Speed3).
- Speed 4 (4rd stage) – The speed is turned ON when the value of the room temperature is higher than the value (T_{set} + Threshold Speed 4 + hysteresis) and turned OFF when the room temperature value reaches the value (T_{set} + Threshold Speed 4)
- Speed 5 (5rd stage) – The speed is turned ON when the value of the room temperature is higher than the value (T_{set} + Threshold Speed 5 + hysteresis) and turned OFF when the room temperature value reaches the value (T_{set} + Threshold Speed 5)

If "Fan level 1-byte data type" is selected as "Enumerated", what fan speed calculated according to above graph, is sent over 1 byte object. For example; If fan speed was calculated as speed 2, 2 is sent over fan speed object.

If "Fan level 1-byte data type" is selected as "Scaling", fan level scaling value is sent according to fan level limits table. For example; if "Fan level 2 threshold value" is 40% and fan speed was calculated as speed 2, %40 value is sent over fan speed object.

4.7.10.2. Fan Proportional Control

Proportional – Integral control (PI control) is explained by the relationship shown below:

$$\boldsymbol{control\ variable(t)\ =\ Kp\ \times\ error(t)}$$

whereby:

$$error(t)\ =\ (Setpoint\ -Measured\ temperature)\ in\ heating$$
$$error(t)\ =\ (Measured\ temperature\ -Setpoint)\ in\ cooling$$
$$Kp\ =\ proportional\ constant$$

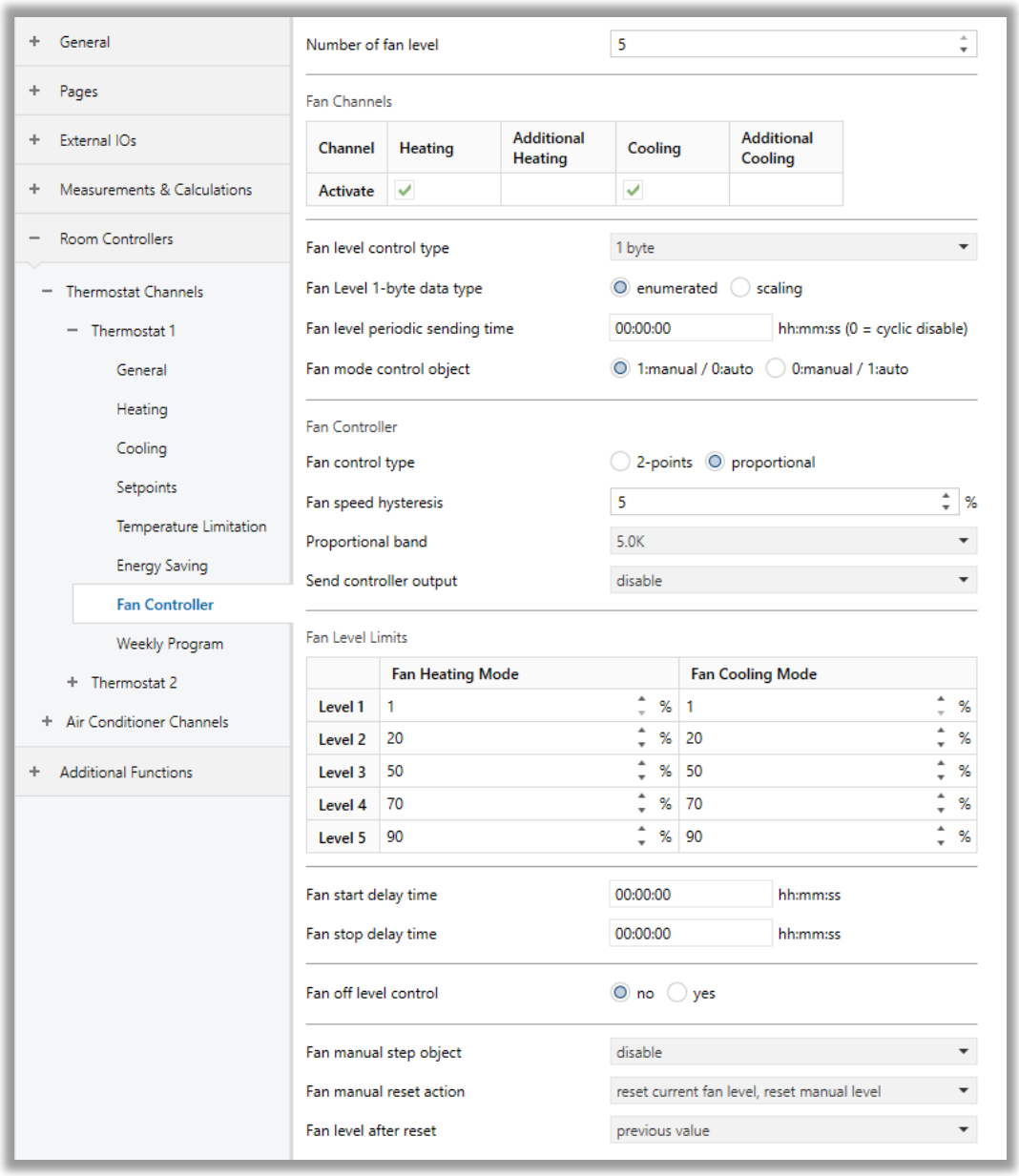


Fig. 64: Fan Controller Proportional Control Configuration

The control variable contains proportional (Kp) constants to eliminate errors. In practice, intuitively generated values are generally used.

$$\boldsymbol{Proportional\ band\ BP\ [K]\ =\ 100\ /\ Kp}$$

The proportional band is the error value that determines the maximum deflection output as 100%.

For example, a regulator with a proportional band of 5 K provides a 100% control output when the Setpoint = 20°C and the measured temperature is ≤ 15°C in heating; in the cooling conduction mode, it provides a 100% control output when the Setpoint = 24°C and the measured temperature is ≥ 29°C. As shown in the figure, a regulator with a small proportional band tends to provide higher values of the control variable for small errors than a regulator with a higher proportional band.

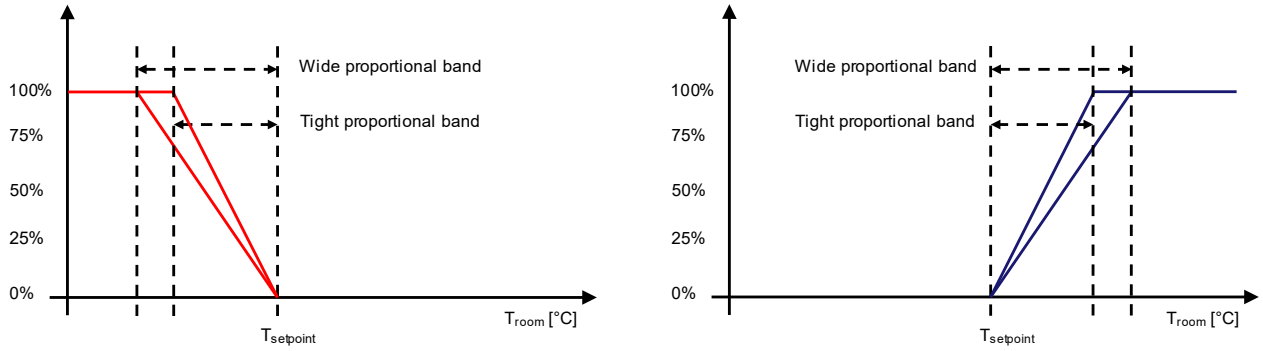


Fig. 65: Fan Controller Proportional Control

The control output is compared to the limit value of fan speed. The fan speed is assigned according to whether the limit values is exceeded or below.

For example, fan level limits are assigned subsequently as 1, 20, 50, 70 and 90 for heating or cooling mode. Assume that the current working mode is Heating and the fan proportional controller generates %65 control value. The control value is compared to fan level limits and as seen the %65 control value is higher than the limits value of levels 1, 2 and 3. So, the fan level is assigned to Level 3.

Note: Fan controller have feedback objects for syncing with controlled device. These objects are not for changing fan level but showing actual value of controlled device. For changing fan level manually manual fan level objects should be used.

4.7.10.3. Parameters List

PARAMETER	DESCRIPTION	VALUES
Number of fan level	The number of fan levels is determined with this parameter.	1…**5**
Channel Heating Activate	This parameter allows the fan controls to work with the heating system. If the heating system is checked, the fan can't connect to the additional heating system at the same time.	**Checked** Unchecked
Channel Additional Heating Activate	This parameter allows the fan controls to work with the additional heating system. If the additional heating system is checked, the fan can't connect to the heating system at the same time.	Checked **Unchecked**
Channel Cooling Activate	This parameter allows the fan controls to work with the cooling system. If the cooling system is checked, the fan can't connect to the additional cooling system at the same time.	**Checked** Unchecked
Channel Additional Cooling Activate	This parameter allows the fan controls to work with the cooling system. If the additional cooling system is checked, the fan can't connect to the cooling system at the same time.	Checked **Unchecked**
Fan level control object	This parameter allows the control of the fan speed with 1-bit individual or 1 byte or 1 bit /1 byte object.	1 bit 1 byte **1 bit / 1 byte**
-> Fan level control data type[1]	This parameter is used to determine with which data type the fan level is sent to the bus. **Enumerated:** 0~5 value is sent. **Scaling:** The percentage equivalent of the fan level value in the fan level limits table.	**Enumerated** Scaling
Fan level periodic sending time	This parameter determines the time of the fan level value to be sent periodically.	**00:00:00**…18:12:15
Fan mode control object	Manual or automatic fan speed control is selected with this parameter.	1: manual / 0: auto **0: manual / 1: auto**
Fan control type	This parameter determines the fan controller type.	**2-points** Proportional
-> Fan speed hysteresis[2]	This parameter determines the fan speed hysteresis value at which switchover to the next fan speed occurs. Using hysteresis avoids continual switching between the fan speeds caused by fluctuating input signals around the limit value.	Values depend on fan controller type

-> Fan Level X Threshold[2]	This parameter determines the fan level X threshold value.	0.5K…5.0K (°C) 0.9K…18.0K (°F)
-> Proportional band[3]	This parameter determines the proportional band of the fan controller.	0.5K… **5K** …10.0K (°C) 0.9K… 9**K** …18.0K (°F)
Fan Heating Mode Level [1…5]	The lower limit value of the 1…5 speed is determined with this parameter.	1…100
Fan Cooling Mode Level X	The lower limit value of the 1…5 speed is determined with this parameter.	1…100
Fan start delay time	This parameter is used to determine the delay time for switching to a higher fan speed than zero.	**00:00:00**…18:12:15
Fan stop delay time	This parameter is used to determine the delay time for switching to zero fan speed.	**00:00:00**…18:12:15
Fan off level control	This parameter is used to enable fan off level control.	**No** Yes
-> Fan off level[4]	This parameter determines the speed of the fan off state.	Values depend on number of fan level.
Fan manual step object	This parameter allows the control of the fan speed with 1 – bit object	**Disable** Increase/decrease (1.007) Up/down (1.008)
Fan manual reset action	This parameter is used to determine what the action is after the value of controller that is connected to fan, is zero in fan manual mode. **No action:** Do nothing, continue to work. **Reset current fan level, hold manual level:** Current manual fan level resets but the previous manual level saves in memory. When the controller value is higher than zero again or manual fan level is changed with the object or thermostat extension of the push button, the manual fan level begins with the value in memory. **Reset current fan level, reset manual level:** Manual fan levels that are current and saved in memory, reset.	No action **Reset current fan level, hold manual level** Reset current fan level, reset manual level
Fan level after reset	The desired fan level after a power failure is determined with this object.	**Previous value** Off Level 1…5 Auto

[1] This parameter is visible when the parameter "Fan level control object" is set to "1 byte" or "1 bit / 1 byte".
[2] This parameter is visible when the parameter "Fan control type" is set to "2-points".
[3] This parameter is visible when the parameter "Fan control type" is set to "Proportional".
[4] This parameter is visible when the parameter "Fan off level control" is set to "Yes".

4.7.11. Thermostat – Weekly Program

Weekly Thermostat Program can be configured over the device. The weekly program works with if HVAC mode is Auto. If HVAC mode is set over object as Auto but the "Thermostat Time" object hasn't been received yet and until the "Thermostat Time" object is received, weekly program doesn't work. During the weekly program runs, the users can change the HVAC mode anytime.

If "Weekly program" parameter is selected as "enable" and "Thermostat Time" object was received, thermostat runs according to weekly program table. If weekly program is active, but any time zone isn't configured, Auto HVAC mode is ended and the HVAC mode switches Comfort mode.

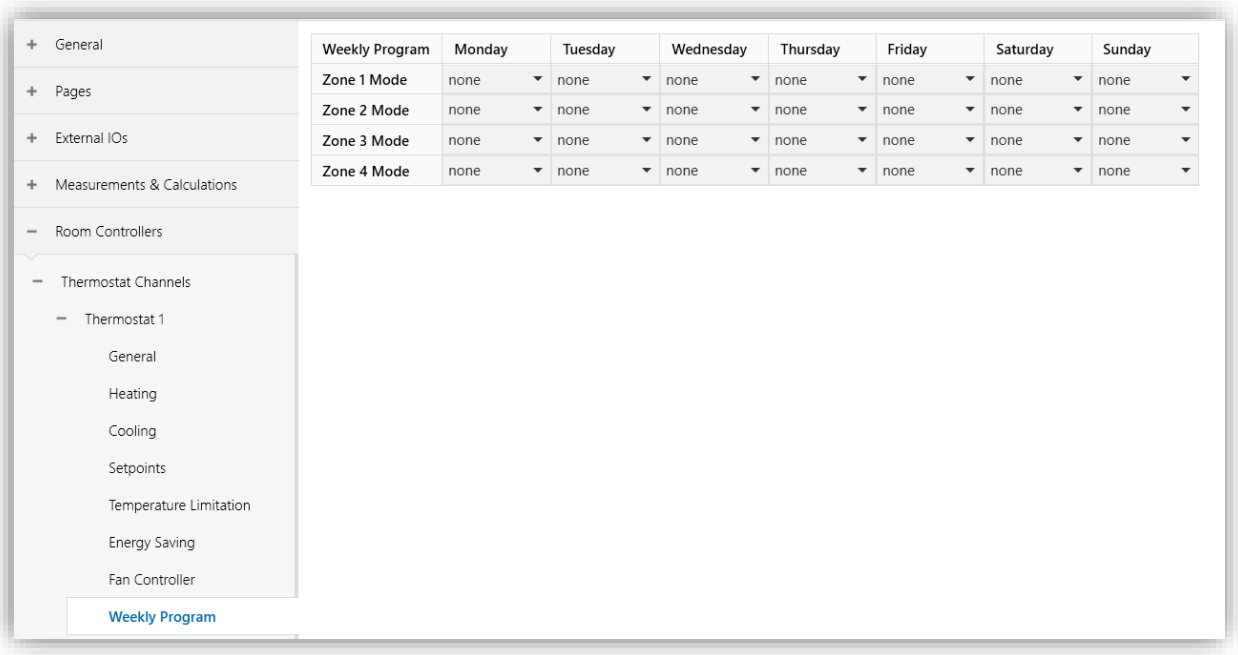


Fig. 66: Weekly Program Configuration

4.7.11.1. Parameters List

PARAMETER	DESCRIPTION	VALUES
Zone X Mode	This parameter is used to determine which HVAC mode will be active according to selected day, hour and minute.	**None** Comfort Standby Economy Protection
=> Zone X Hour	This parameter is used to determine the hour that the HVAC mode will be active.	**0** … 23
=> Zone X Minute	This parameter is used to determine the minute that the HVAC mode will be active.	**0** … 59
Auto switch-over HVAC modes	If this parameter is enabled, HVAC mode is changed according to the weekly program table.	**Disable** Enable

4.8.11. Thermostat – Slave

Thermostat can operate in slave mode. It can work in compatibility with master room controllers.
However, some configurations must be the same as the master device to work properly. Such as available HVAC modes, heating/cooling control modes, fan level settings etc.

The slave device can control the heating/cooling mode, fan level, fan mode, HVAC mode, setpoint etc. The controller is just in the master. The slave device is only for indicating the room controller values and sending some commands.

In slave mode, the setpoint temperature range that can be set from the thermostat control page is between **10 and 40 °C**.

4.7.11.1. Parameters List

PARAMETER	DESCRIPTION	VALUES
Temperature source	This parameter determines the source of room temperature to be displayed.	**Internal temperature** Temperature object Calculation 1…6
Master controller mode	This parameter determines the controller mode of master device. This parameter must be the same as master device to work properly.	**Heating** Cooling Heating/Cooling
Temp Unit	This parameter determines the temperature unit of the room and the temperature setpoint to be displayed.	**Celsius** Fahrenheit
Manual setpoint type	This parameter determines the setpoint temperature type to be displayed.	**Individual** Dependent
Manual setpoint step	This parameter determines the setpoint step to be sent.	0.1K … **0.5K** … 3.5K (˚C) 0.18K … **0.9K** … 6.3K (˚F)
Comfort Mode Activate	This parameter is used to determine the activation of comfort mode. If this parameter is checked, comfort mode can be useable.	Unchecked **Checked**
Standby Mode Activate	This parameter is used to determine the activation of standby mode. If this parameter is checked, standby mode can be useable.	Unchecked **Checked**

Economy Mode Activate	This parameter is used to determine the activation of economy mode. If this parameter is checked, economy mode can be useable.	Unchecked **Checked**
Protection Mode Activate	This parameter is used to determine the activation of protection mode. If this parameter is checked, protection mode can be useable.	Unchecked **Checked**
Fan indicator used for master control	This parameter determines that the fan controller is used for fan controller indicator in slave device. The fan controller parameter must be the same as master device.	**No** Yes
-> Number of fan level[1]	This parameter determines the maximum fan speed to be displayed. This parameter must be the same as master device.	1…**5**
-> Fan level control type[1]	This parameter determines object data type of fan speed. This parameter must be the same as master device.	1-bit **1-byte** 1-bit/1-btyte
-> Fan level 1-byte data type[2]	This parameter is used to determine with which data type the fan level is sent to the bus. **Enumerated:** 0~5 value is sent. **Scaling:** The percentage equivalent of the fan level value in the fan level limits table. This parameter must be the same as master device.	**Enumerated** Scaling
-> Fan mode control object[1]	This parameter determines which data is received to switch between fan modes. This parameter must be the same as master device.	**1: manual / 0: auto** 0: manual / 1: auto
-> Fan level X limits – Heating Mode[3]	The lower limit value of the 1…5 speed is determined with this parameter.	%0…%100
-> Fan level X limits – Cooling Mode[3]	The lower limit value of the 1…5 speed is determined with this parameter.	%0…%100

[1] This parameter is visible when the parameter "Fan indicator used for master control" is set to "Enable

[2] This parameter is visible when the parameter "Fan level control object" is set to "1 byte" or "1 bit / 1 byte".

[3] This parameter is visible when the parameter "Fan Level 1-byte data type" is set to "Scaling".

4.8. Room Controller – Air Conditioner

4.8.1. Air Conditioner X

All configurations related to air conditioner control on the iX3 are described in the sections of this chapter. This parameter page will be shown when it is enabled in the "Air Conditioner X" parameter page section. The users can control various features via AC controller.

- AC mode settings
- External AC mode settings
- AC fan settings
- AC fan direction (vertical/horizontal) settings
- Error information up to last 8 errors. (Show on AC control page)

These features can be controlled via AC control page in iX3 device. All features can be configured over ETS software. Also, iX3 AC controller can work compatible with INTERRA AC Gateway products.

AC controller room temperature source can be internal temperature sensor or external sensor via group object. The users can be set the setpoint min/max range over ETS software.

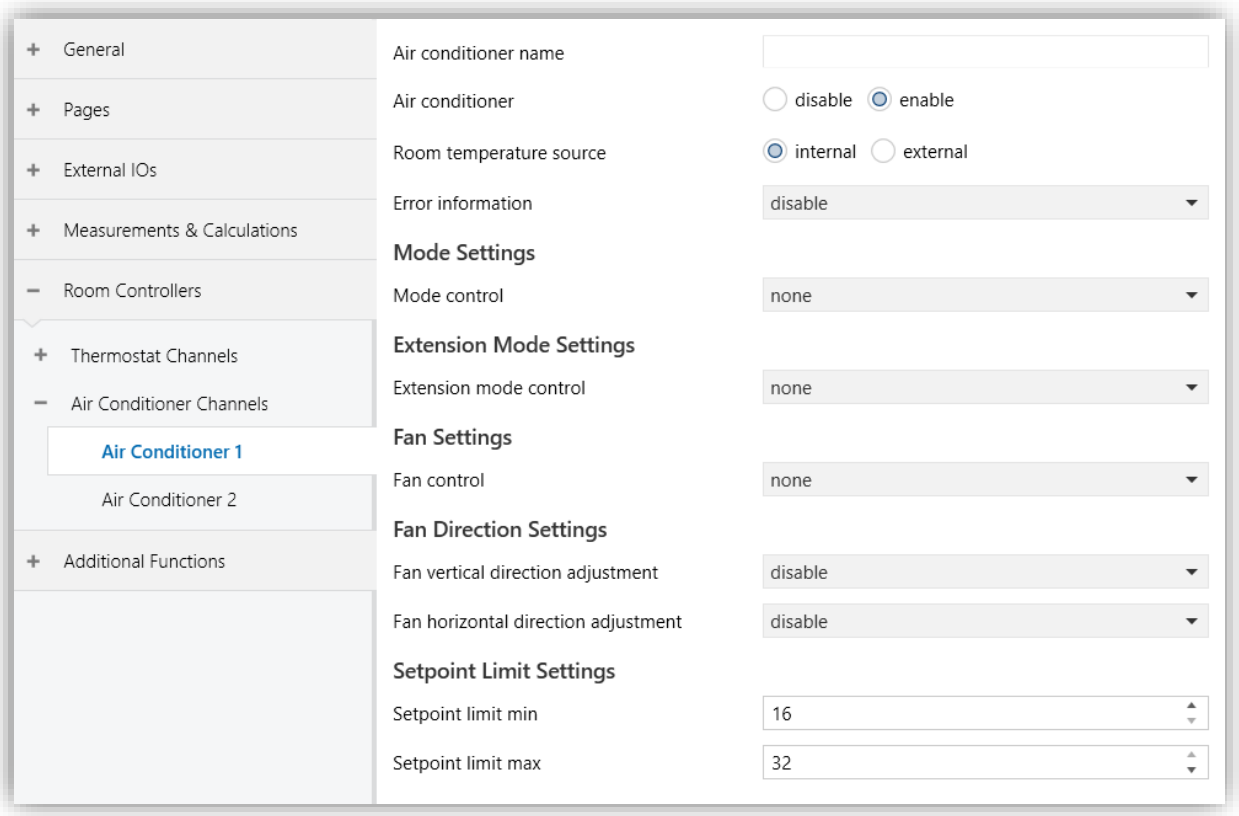


Fig. 67: Air Conditioner X Configuration

4.8.1.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Air conditioner name	This parameter is used to type an Air Conditioner name. The name can be consisting of 40 characters.	**40 Bytes allowed**
Air conditioner	This parameter is used to control the air conditioner features.	Disable **Enable**
Room temperature source	This parameter determines the source of room temperature. **Internal:** Use internal temperature sensor **External:** Use external temperature sensor via group object.	**Internal** External
Error Information	This parameter determines the object data type of error information. AC device is sent an error by these group objects. Received errors are shown on AC control pages.	**Disable** 2-bytes value 14-bytes text Both

4.8.2. Air Conditioner – Mode Settings

The users can control AC fan mode over DPT 20.105 group object values. The output value of related control mode can be configured. Also, the status of each mode, the device will update the icon status according to the feedback value received.

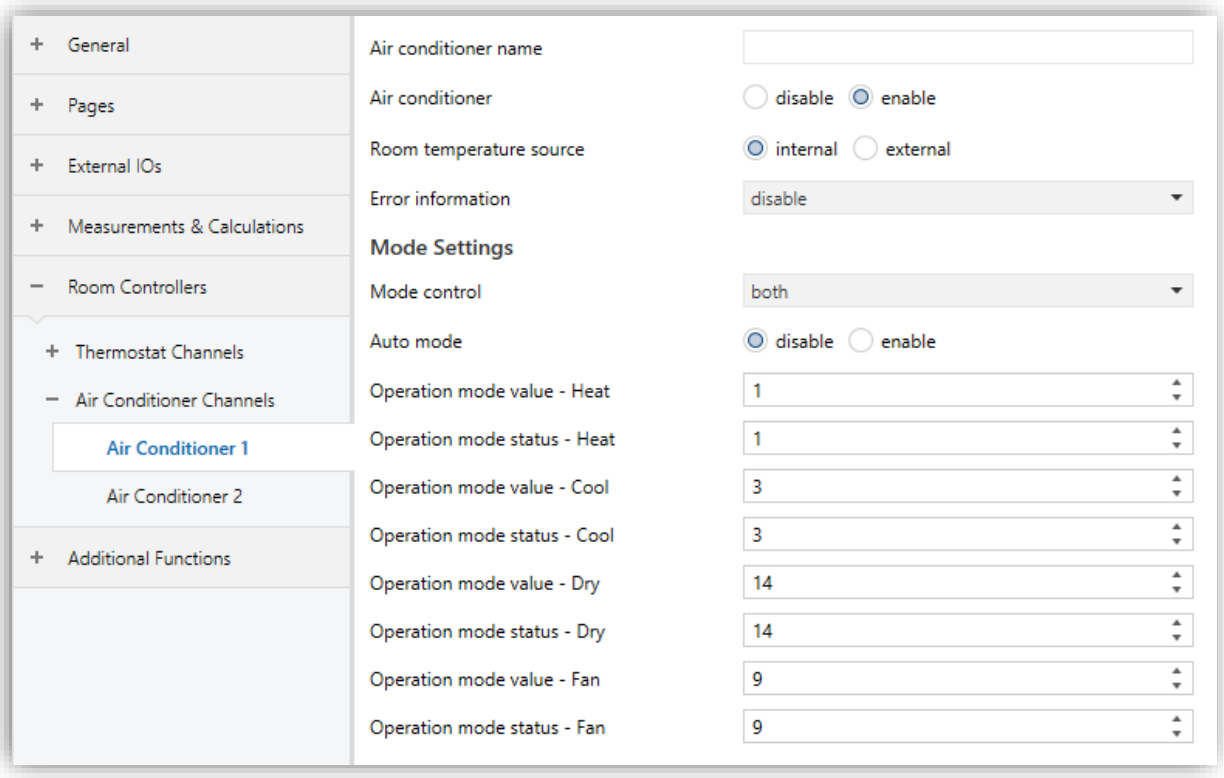


Fig. 68: Air Conditioner X – Mode Settings Configuration Page

4.8.2.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Mode control	This parameter determines the object data type of AC device's control mode.	**None** Bit Byte Both
-> Auto mode[1]	This parameter determines that auto mode is available in AC's control mode.	**Disable** Enable
-> Operation mode value – Auto[2]	This parameter determines the output value of the auto mode.	**0**…255
-> Operation mode status – Auto[2]	This parameter determines the status value of the auto mode. The device will update the icon according to this value.	**0**…255
-> Operation mode status – Heat[2]	This parameter determines the output value of the heat mode.	0…**1**…255
-> Operation mode status – Heat[2]	This parameter determines the status value of the heat mode. The device will update the icon according to this value.	0…**1**…255
-> Operation mode status – Cool[2]	This parameter determines the output value of the cool mode.	0…**3**…255
-> Operation mode status – Cool[2]	This parameter determines the status value of the cool mode. The device will update the icon according to this value.	0…**3**…255
-> Operation mode status – Dry[2]	This parameter determines the output value of the dry mode.	0…**14**…255
-> Operation mode status – Dry[2]	This parameter determines the status value of the dry mode. The device will update the icon according to this value.	0…**14**…255
-> Operation mode status – Fan[2]	This parameter determines the output value of the fan mode.	0…**9**…255
-> Operation mode status – Fan[2]	This parameter determines the status value of the fan mode. The device will update the icon according to this value.	0…**9**…255

[1] This parameter is visible when the parameter "Mode Control" is set to "Bit" or "Both".
[2] This parameter is visible when the parameter "Mode Control" is set to "Byte" or "Both".

4.8.3. Air Conditioner – Extension Mode Settings

Except the common control mode, iX3 AC controller provides up to 7 external mode settings for different control modes. The users can configure the extension modes according to AC modes. The users can configure the extension mode's names up to 7 characters.

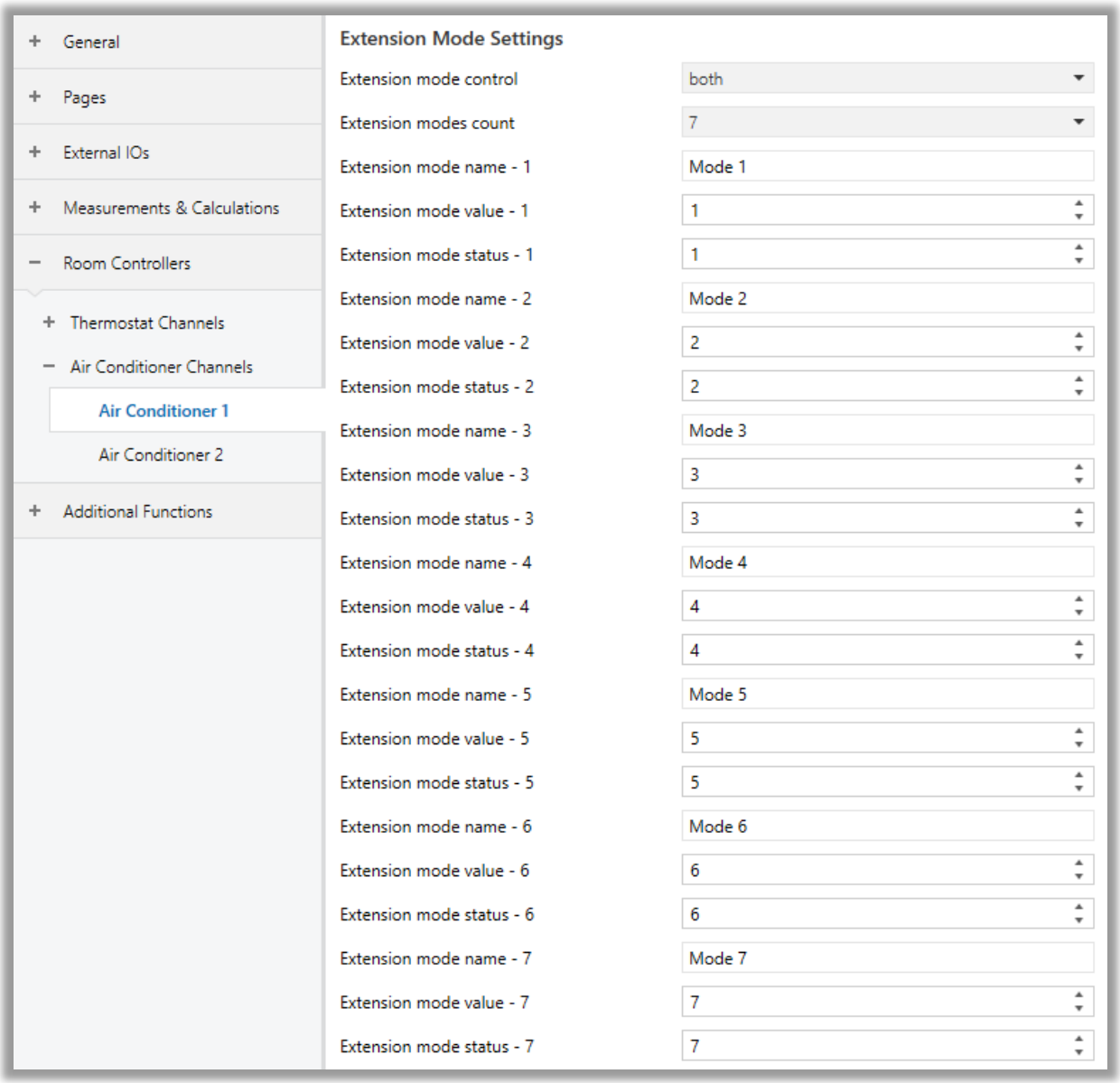


Fig. 69: Air Conditioner – Extension Mode Settings Configuration Pages

4.8.3.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Extension mode control	This parameter determines the object data type of extension mode's output value.	**None** Bit Byte Both
-> Extension modes count[1]	This parameter determines the count of extension modes.	**0**…7
-> Extension mode name – X[1]	This parameter determines the name of the extension mode. The name is shown on the AC control pages.	**Up to 7 characters (UTF-8)**
Extension mode value – X[2]	This parameter determines the output value of extension mode.	**0…**255
Extension mode status – X[2]	This parameter determines the status value of the extension mode. The device will update the icon according to this value.	**0…**255

[1] This parameter is visible when the parameter "Extension mode control" is set to "Bit" or "Byte" or "Both".

[2] This parameter is visible when the parameter "Extension mode control" is set to "Byte" or "Both".

4.8.4. Air Conditioner – Fan Settings

The user can control up to 5 level fan speed via AC controller. The output value of each level can be configured over ETS software. Also, the status of each mode, the device will update the icon status according to the feedback value received.

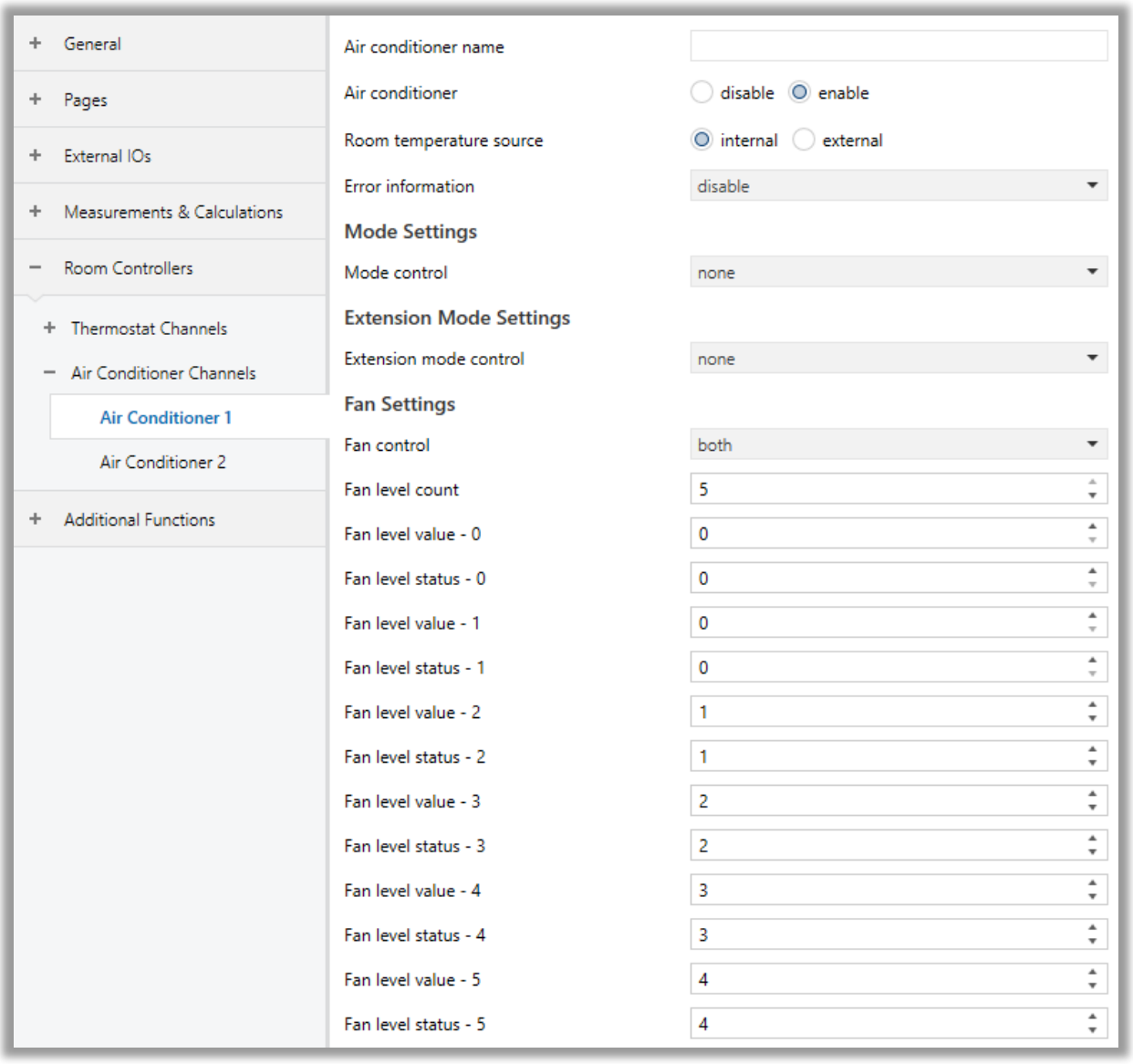


Fig. 70: Air Conditioner – Fan Settings Configuration Page

4.8.4.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Fan control	This parameter determines the object data type of fan level's output value.	**None** Bit Byte Both
-> Fan level count[1]	This parameter determines the maximum level of fan speed.	1…**5**
-> Fan level value – X[2]	This parameter determines the output value of fan level X.	**0…**255
-> Fan level status – X[2]	This parameter determines the status value of the fan level X. The device will update the icon according to this value.	**0…**255

[1] This parameter is visible when the parameter "Fan control" is set to "bit" or "byte" or "both".
[2] This parameter is visible when the parameter "Fan control" is set to "byte" or "both".

4.8.5. Air Conditioner – Fan Direction Settings

The users can control horizontal and vertical swings both fixed and moving separately. Up to 5 levels can be controlled. The output value of each level can be configured via ETS software.

Fan level 0 stands auto mode in AC controller. If AC fan level is auto, fan level icon is not shown on the screen, just auto mode icon is shown.

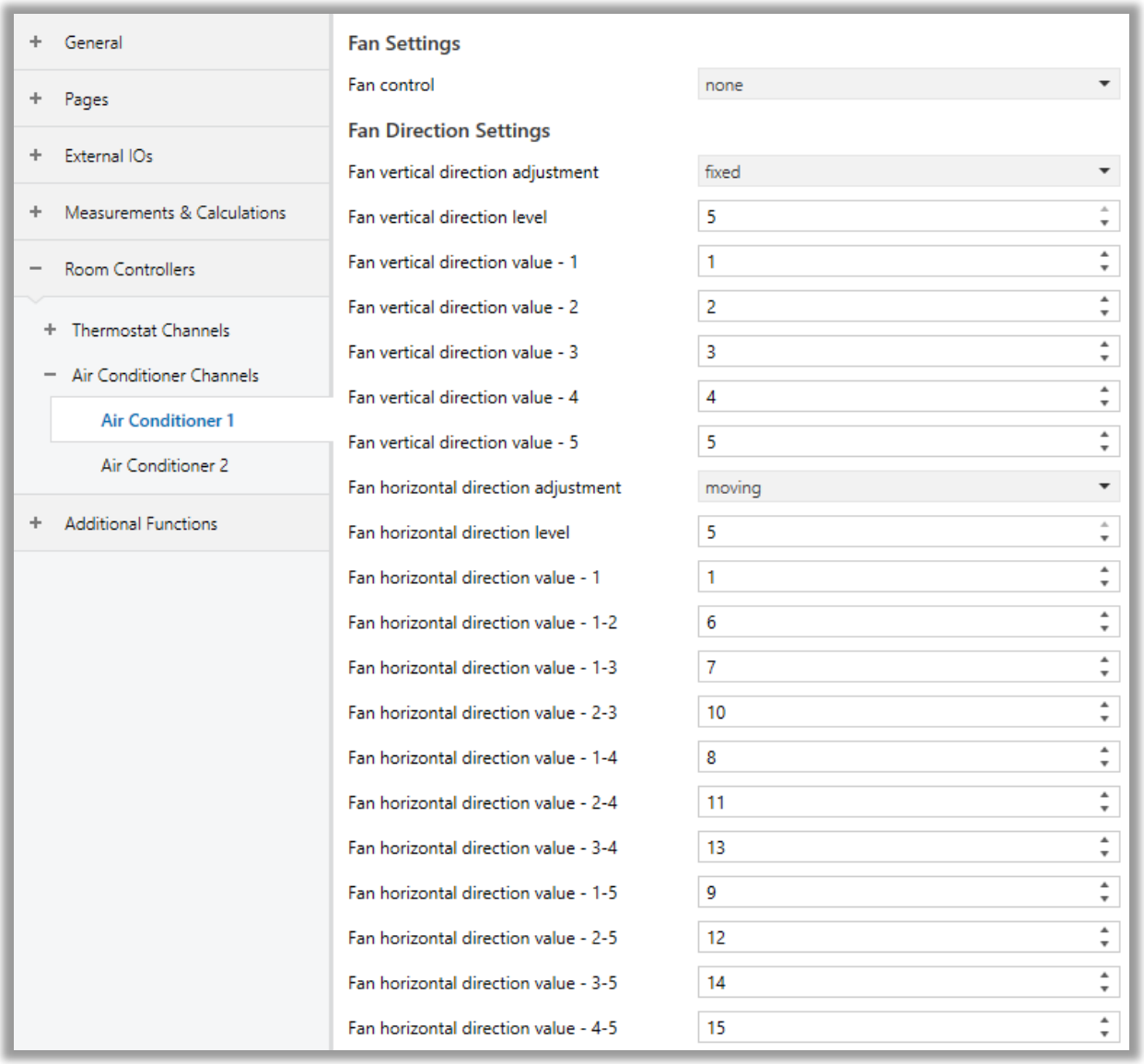


Fig. 71: Air Conditioner – Fan Direction Settings Configuration Page

4.8.5.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
-> Fan vertical direction adjustment	This parameter determines the control type of vertical swing. **Fixed:** Stop fixed level. **Moving:** Moving between 2 level.	**Disable** Fixed Moving Both
-> Fan vertical direction level[1]	This parameter determines the maximum swing level of vertical direction.	0**…5**
-> Fan vertical direction value – X[1]	This parameter determines the output value of horizontal direction level.	**0…**255
-> Fan horizontal adjustment	This parameter determines the control type of horizontal swing. **Fixed:** Stop fixed level. **Moving:** Moving between 2 level.	**Disable** Fixed Moving Both
-> Fan horizontal direction level[2]	This parameter determines the maximum swing level of horizontal direction.	0**…5**
-> Fan horizontal direction value – X[2]	This parameter determines the output value of horizontal direction level.	**0…**255

[1] This parameter is visible when the parameter "Fan vertical direction adjustment" is **not** set to "None".

[2] This parameter is visible when the parameter "Fan horizontal direction adjustment" is **not** set to "None".

4.8.6. Air Conditioner – Setpoint Limit Settings

The users can control the setpoint temperature over AC control page. Min/max. setpoint range can be configured over ETS software.

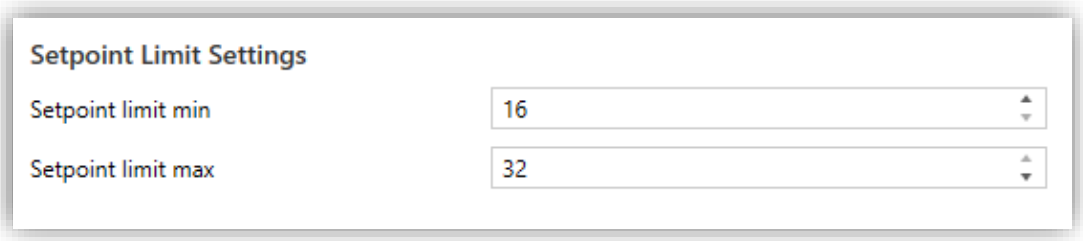


Fig. 72: Air Conditioner – Setpoint Limit Settings Configuration Page

4.8.6.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Setpoint limit min	This parameter determines the minimum value of setpoint temperature.	**16**…32
Setpoint limit max	This parameter determines the maximum value of setpoint temperature.	16…**32**

4.11. Additional Functions – Logics

This section describes the logical function modules of the iX3. With the logical function blocks on iX3, a logical expression can be created with the data coming through the local digital inputs or external inputs, and various 'TRUE' or 'FALSE' results can be obtained. actions can be taken and scenarios can be triggered.

4.11.1. Logics – General

This section describes the general parameters of the logical association module of iX3. Parameters must be configured separately for each logic block.

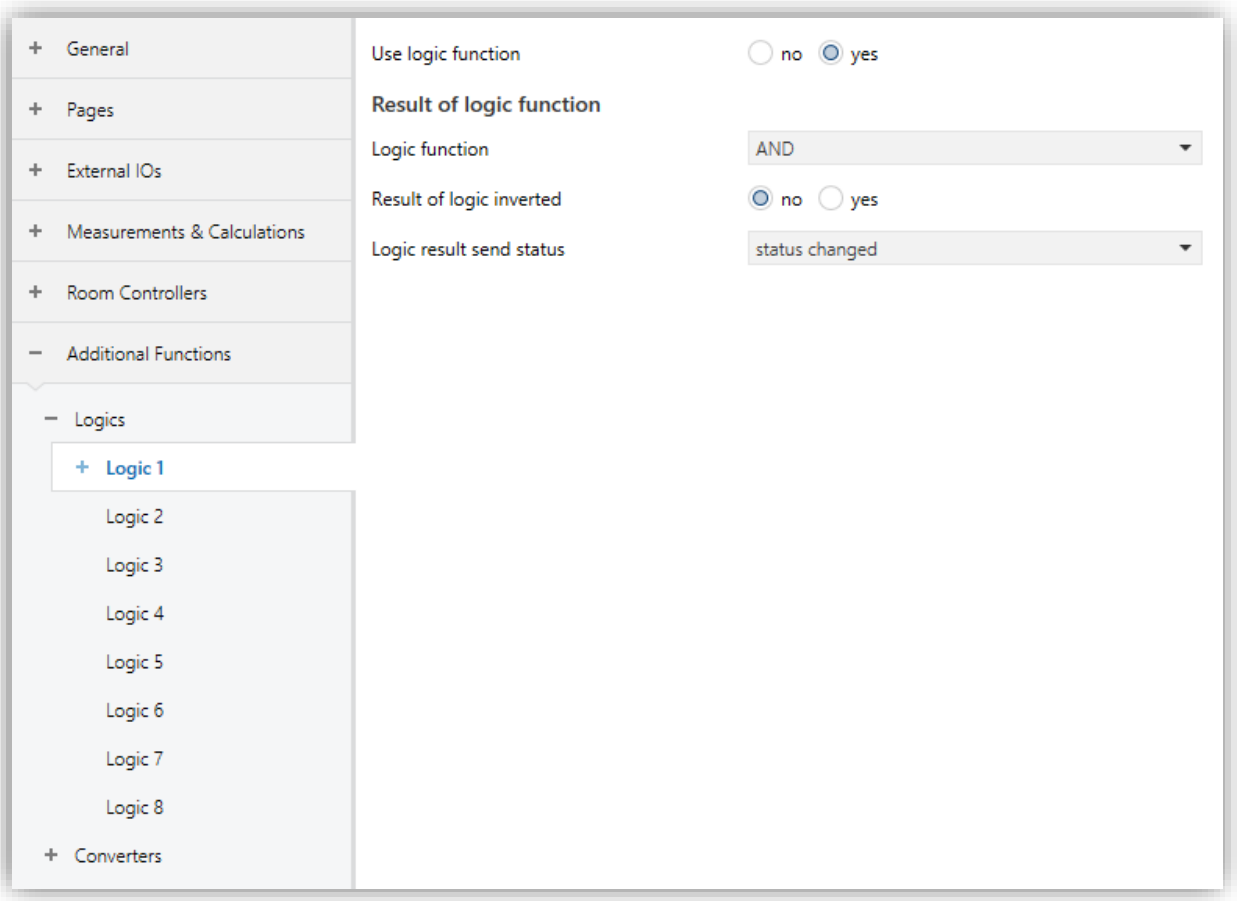


Fig. 75: Logics – General Configuration Page

4.11.1.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Use Logic Function	This parameter is used to enable or disable the related logic function gate.	**No** Yes
Logic Function	This parameter is used to determine the logical relation of the parameterized logic inputs. AND**:** All inputs are put into the 'AND' operation. OR**:** All inputs are put into the 'OR' operation. XOR**:** All inputs are put into the 'XOR' operation.	**AND** OR XOR
Result of Logic Inverted	This parameter is used to invert or not invert the calculated logic function block. If it is selected as yes for example, when the logic function gate output is 'TRUE', the output will be 'FALSE'. Vice versa also applies.	**No** Yes
Logic result send status	This parameter is used to determine the logic function block result sending status to the KNX bus.	**Status changed** Status is TRUE Status is FALSE Status changed and periodically Status is TRUE periodically Status is FALSE periodically

4.11.2. Logics – Internal Inputs

This section describes the internal input parameters of the logical association module of iX3. Parameters must be configured separately for each logic block. The source of the internal inputs is "External IOs" channels. If the "External Input X" channel is configured as digital input, this input can be used in logic module.

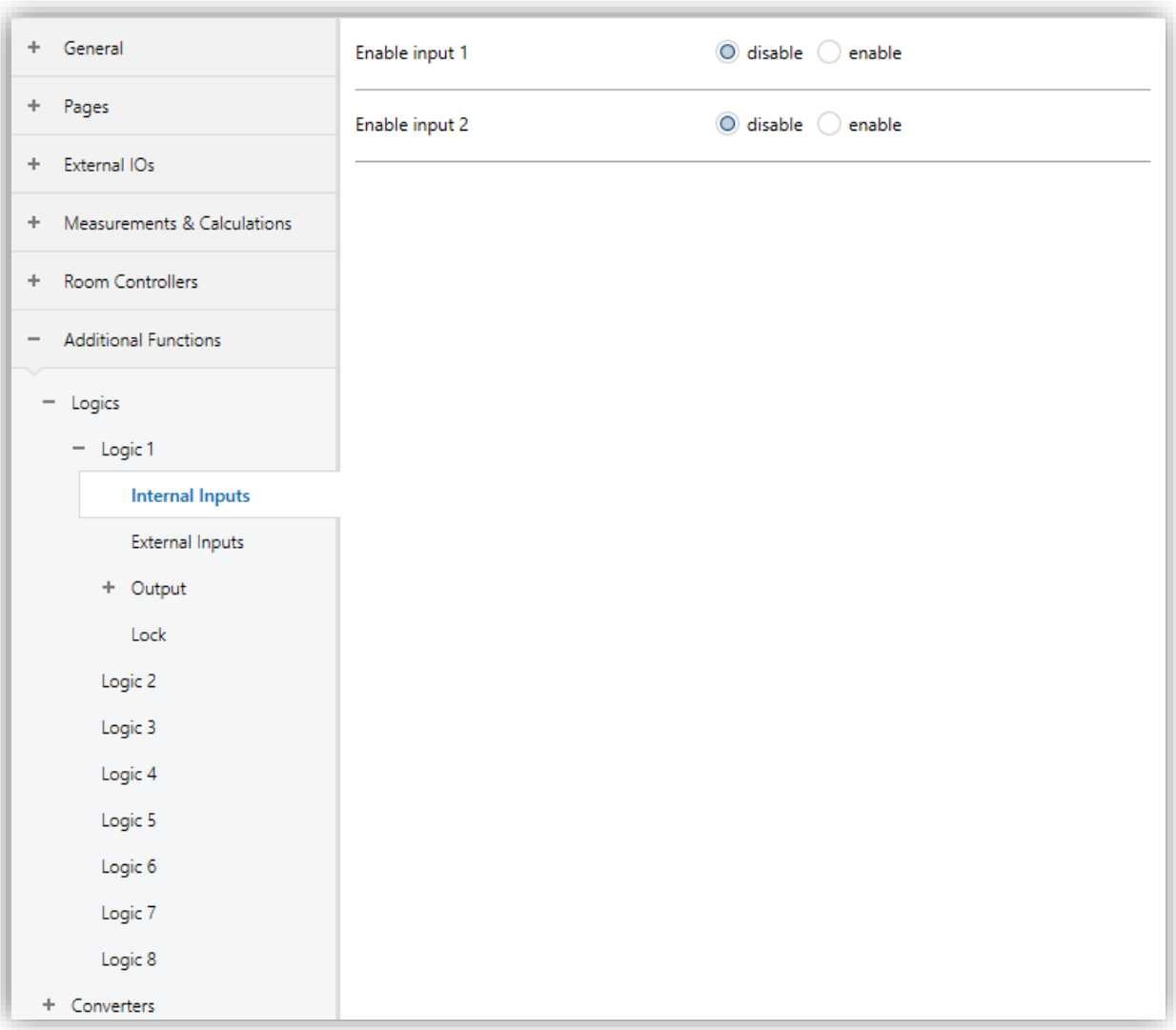


Fig. 76: Logics – Internal Input Configuration Page

4.11.2.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Enable input X	This parameter is used to enable or disable internal input X for logic function block as input.	**Disable** Enable
-> Contact input status	This parameter is used to determine the input status as TRUE or FALSE according to the value. (This is visible if the input is selected as 1 bit)	**'1' is TRUE, '0' is FALSE** '1' is FALSE, '0' is TRUE

4.11.3. Logics – External Inputs

This section describes the external input parameters of the logical association module of iX3. Parameters must be configured separately for each logic block. 3 external digital and 3 external sensor inputs can be processed logically.

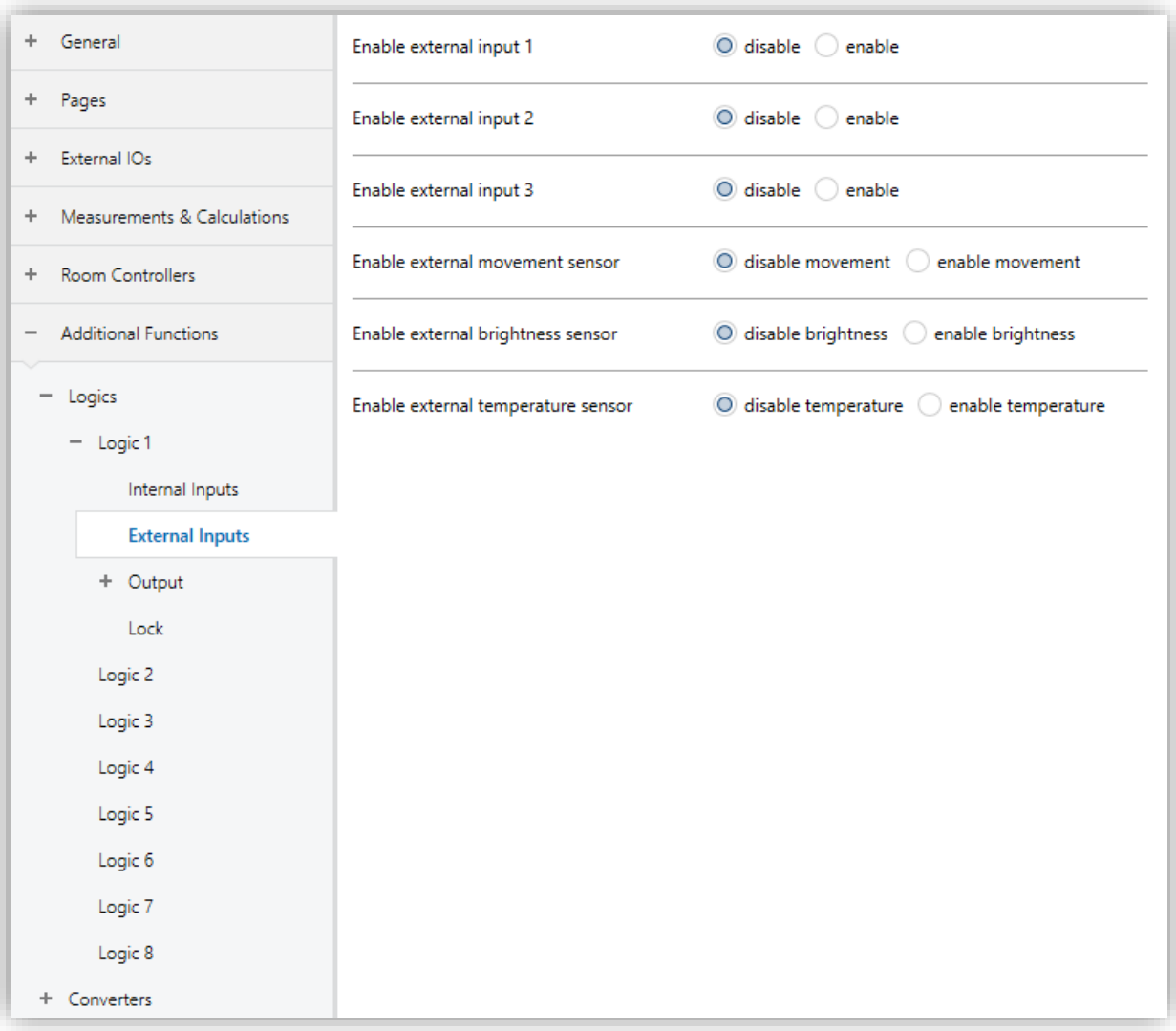


Fig. 77: Logics – External Inputs Configuration

4.11.3.1 Parameters List

PARAMETERS	DESCRIPTION	VALUES
Enable external input X	This parameter is used to enable or disable external input X for logic function block as input.	**Disable** Enable
-> External input type	This parameter is used to determine the external input type of the enabled input 1 object.	**1-bit value ('1'/'0')** 1-byte value (0...255) 2-byte threshold (0 …65535) 2-byte float threshold (-50C …100C) 4-byte threshold (0…4294967295)
-> External input status	This parameter is used to determine the input status as TRUE or FALSE according to the value. (This is visible if the input is selected as 1 bit)	**'1' is TRUE, '0' is FALSE** '1' is FALSE, '0' is TRUE
-> External Input value	This parameter is used to determine the external input threshold value to evaluate the input status as TRUE or FALSE.	0…**100**…255 0…**1000**…65535 -500…**0**…1000 0…**10000**…4294967295
-> External input status	This parameter is used to determine the input status as TRUE or FALSE according to the value. (This is visible if the input is not selected as 1 bit)	TRUE if input value >= threshold else FALSE TRUE if input value <= threshold else FALSE
Enable Movement Sensor	This parameter is used to enable or disable the movement sensor: External movement: The external movement information will be used for movement detection.	**Disable movement** External movement
-> Internal Movement Sensor Status	This parameter is used to determine when the internal movement sensor detects a movement is accounted as TRUE or FALSE.	Movement sensor detected is FALSE else is TRUE **Movement sensor detected is TRUE else is FALSE**
Enable Brightness Sensor	This parameter is used to enable or disable the brightness sensor.	**Disable Brightness** External Brightness

	External Brightness: The external brightness sensor will be used as brightness logic input.	
-> Threshold brightness lower	This parameter is used to determine the lower threshold brightness value.	1…**100**…1200
-> Threshold brightness upper	This parameter is used to determine the upper threshold brightness value.	1…**300**…1200
-> Brightness status	This parameter is used to determine when the ambient brightness value is accounted as TRUE or FALSE.	**In range is TRUE, else FALSE** Out range is TRUE, else FALSE Under lower is TRUE, above upper is FALSE Under lower is FALSE, above upper is TRUE
-> Change brightness threshold via bus	This parameter is used to change the brightness threshold value via a KNX bus object.	**No** Yes
Enable Temperature Sensor	This parameter is used to enable or disable the temperature sensor. KNX temperature: The external temperature sensor will be used as temperature logic input.	**Disable Temperature** External temperature
-> Threshold temperature upper	This parameter is used to determine the lower threshold temperature value.	**-**300…**260**…700
-> Threshold temperature lower	This parameter is used to determine the upper threshold temperature value.	-300…**220**…700
-> Temperature status	This parameter is used to determine when the ambient temperature value is accounted as TRUE or FALSE.	**In range is TRUE, else FALSE** Out range is TRUE, else FALSE Under lower is TRUE, above upper is FALSE Under lower is FALSE, above upper is TRUE
-> Change temperature threshold via bus	This parameter is used to change the temperature threshold value via a KNX bus object.	**No** Yes

4.11.4. Logics – Output

This section describes the general parameters of the logic output functions. The property of each respective output channel is set by configuring the parameters in this section. Also, repetitive sending of output values can be set here.

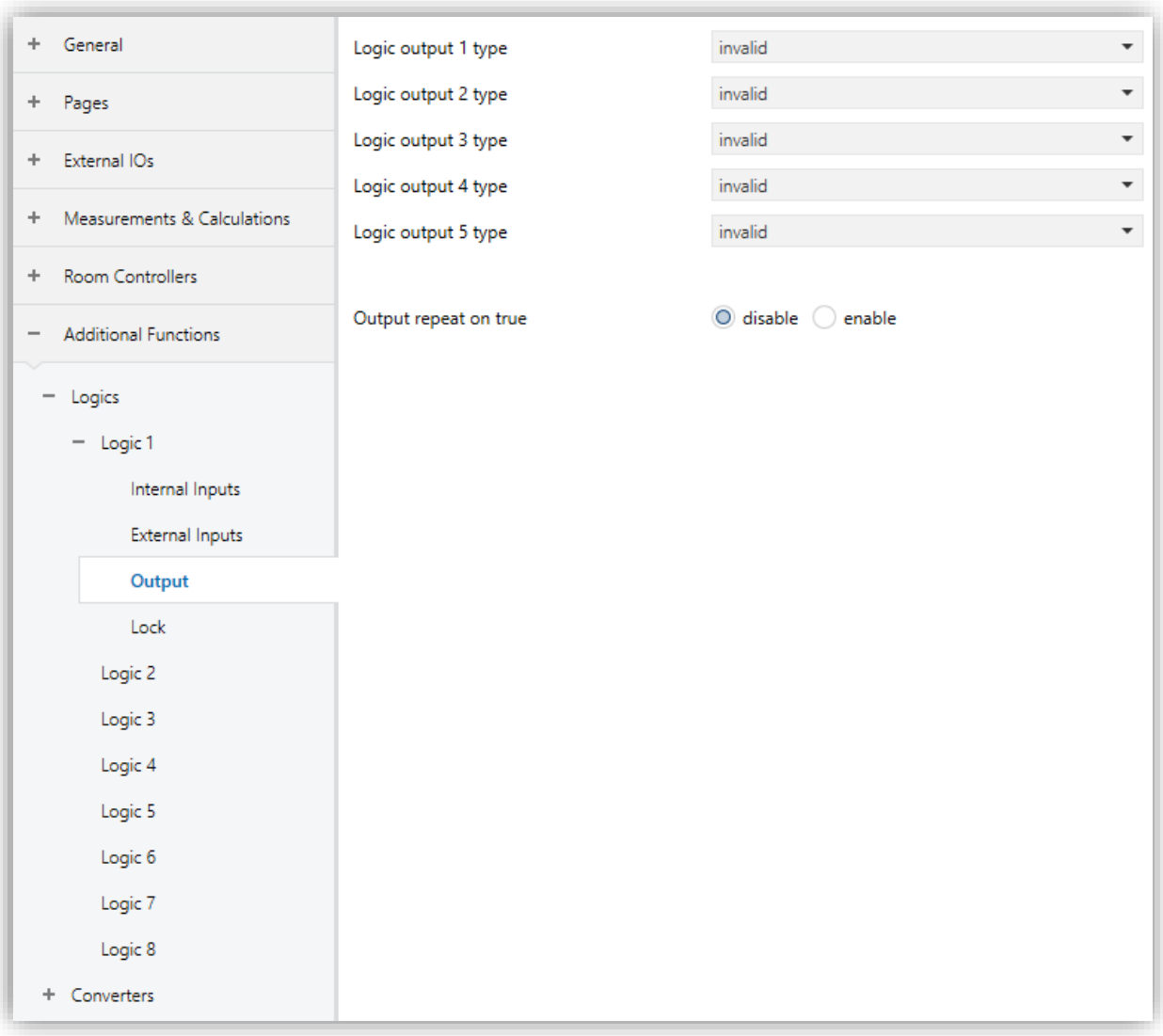


Fig. 78: Logics – Output Configuration

4.11.4.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Logic Output X type (1...5)	This parameter is used to specify the related logic output x channel functionality. If this parameter is selected as invalid, the related output channel will not be used. Other selected options will be configured separately.	**Invalid** Switch controller Absolute dimming controller Shutter controller Alarm controller Percentage control. Sequence control. Scene controller String controller Threshold controller
Output repeat on true	This parameter is used to enable or disable the output repeating time for all output channels when the logic gate state is true.	**On telegram** Off telegram
-> Repeated time interval	This parameter is used to determine the repeated time for all enabled output channels to send output channel values when the logic gate state is true.	0…**120**…65535

4.11.5. Logics – Output 1-5

This section describes parameter configurations for each logic output channel. Although the working principle is the same for all output channels, only the type of values to be sent changes depending on the selected output functionality. For this reason, parameters are described in a common table about only one feature.

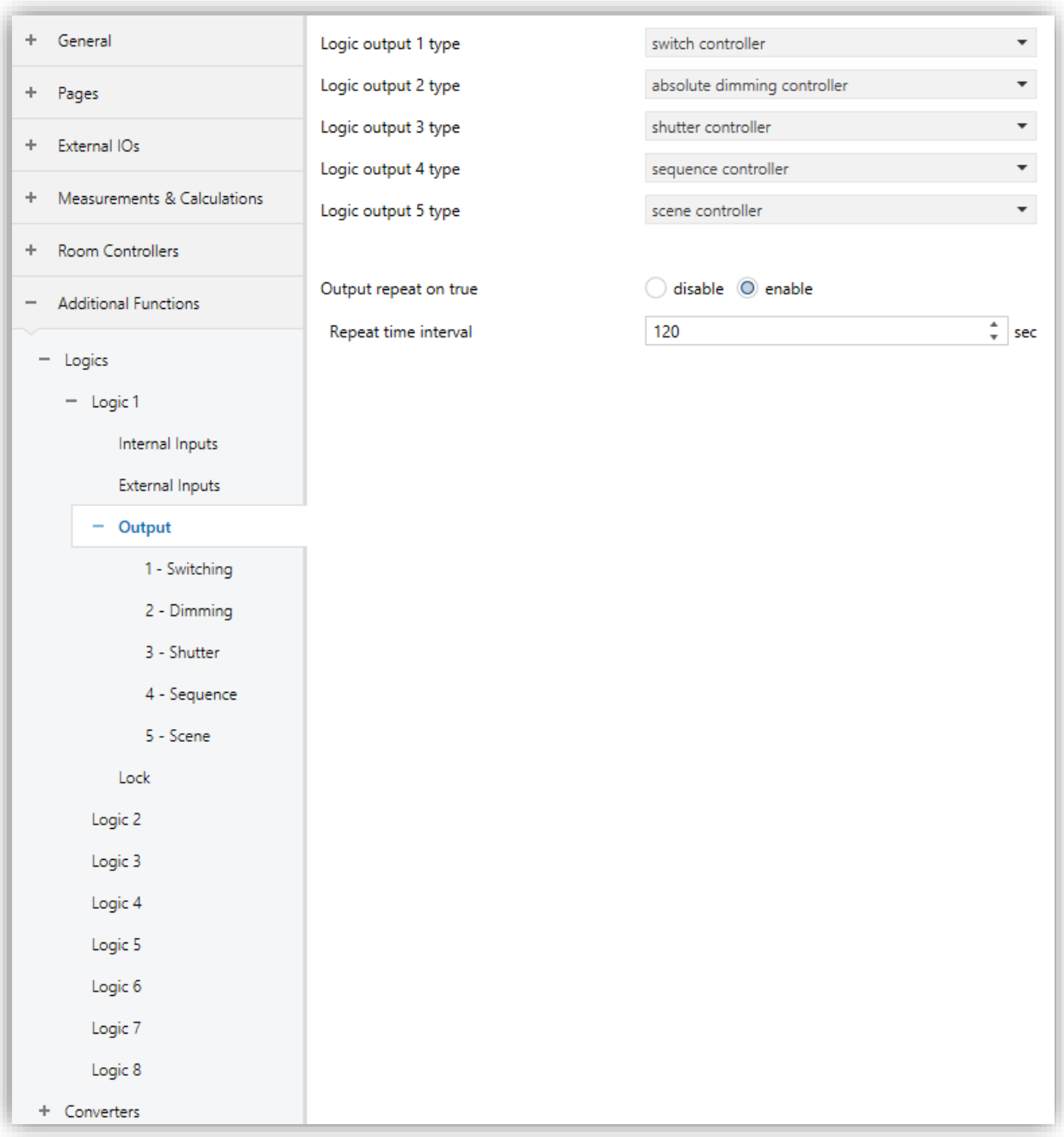


Fig. 79: Logics – Output 1-5 Configuration

4.11.5.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
The status after bus voltage recovery	This parameter is used to determine the logic output channel x status after bus voltage recovery.	**Invalid** Recovery
-> Recovery Defined Value	This parameter is used to determine the output channel x value when the bus voltage has been recovered.	**On**…Off **%0**…%100 **Up**…Down **No alarm**…alarm **Stop**…start **Scene No. 1…64** 14 bytes string 0…**1000**…65535
Send output object when TRUE	This parameter is used to enable or disable the sending output object when the logic gate is true.	No Yes
-> Defined Output Value	This parameter is used to determine the logic output channel x defined value when the logic gate is true.	**On**…Off **%0**…%100 **Up**…Down **No alarm**…alarm **Stop**…start **Scene No. 1…64** 14 bytes string 0…**1000**…65535
-> On Delay Time	This parameter is used to determine the on-delay time of the related logic output channel x when the logic gate is true.	**00:00:00**…18:12:15
-> Change on Time Via Bus	This parameter is used to enable or disable the on-delay time object for changing the delay time on the true state.	**No** Yes
Send output object when FALSE	This parameter is used to enable or disable the sending output object when the logic gate is false.	**No** Yes
-> Defined Output Value	This parameter is used to determine the logic output channel x defined value when the logic gate is false.	**On**…Off **%0**…%100 **Up**…Down **No alarm**…alarm **Stop**…start **Scene No. 1…64** 14 bytes string 0…**1000**…65535

-> On Delay Time	This parameter is used to determine the on-delay time of the related logic output channel x when the logic gate is false.	**00:00:00**…18:12:15
-> Change on Time Via Bus	This parameter is used to enable or disable the on-delay time object for changing the delay time on the false state.	**No** Yes

4.11.6. Logics – Lock

In this section, the locking feature of the logic functions is mentioned. The locking feature is for each logic function gate and is configured separately. Since there are 5 different logic function gates in iX3, a separate configuration is required for each. Since the parameter page for each section is the same, only 1 is explained in this section.

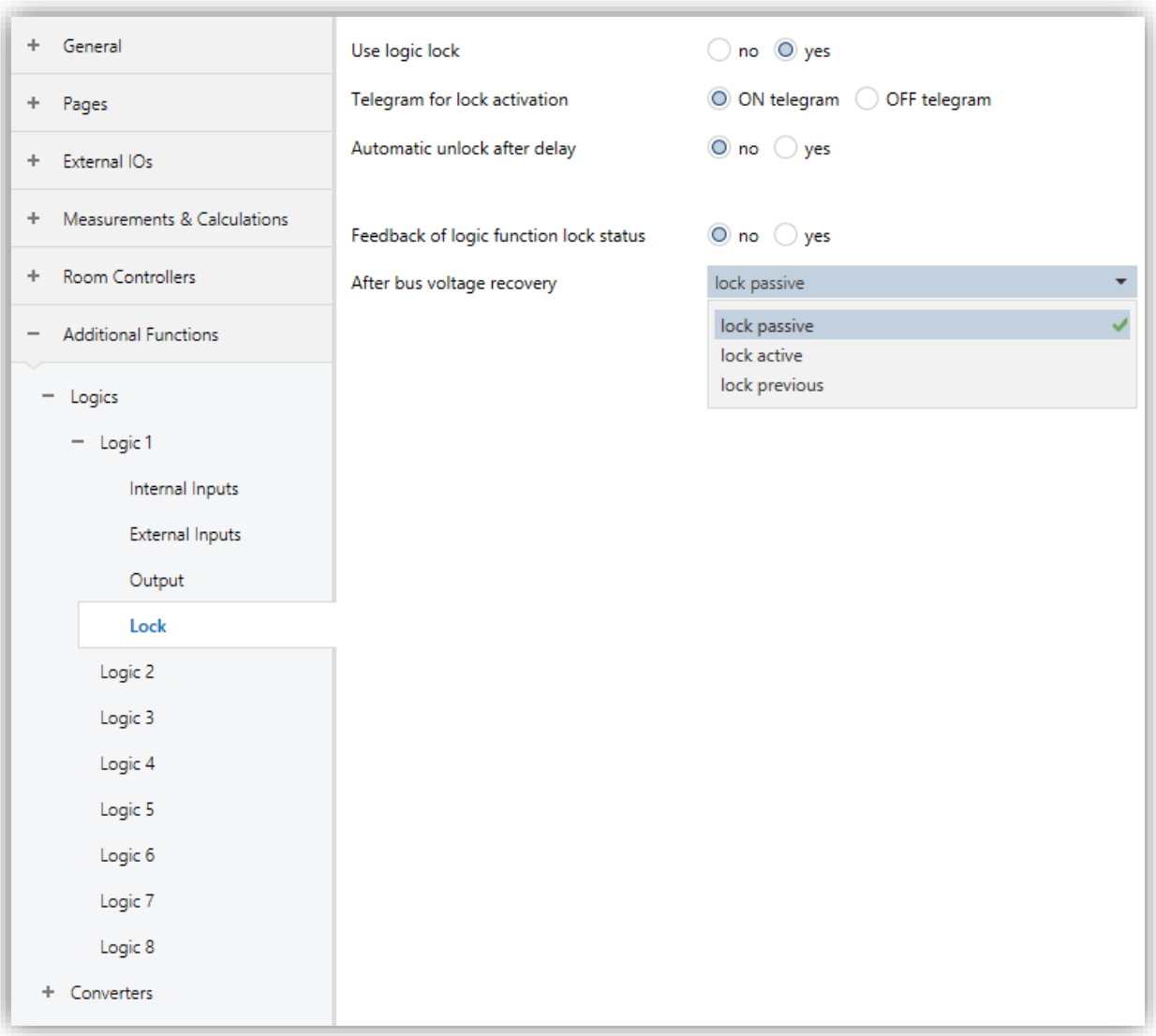


Fig. 80: Logics – Lock Configuration

4.11.6.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Use Logic Lock	This parameter is used to lock the related logic function gate.	**No** Yes
Telegram for Lock Activation	This parameter is used to determine the telegram value that locks the related logic function gate.	**On telegram** Off telegram
Automatic Unlock After Delay	This parameter is used to enable or disable the automatic unlock to unlock the logic gate after a while.	**No** Yes
Automatic unlock time	This parameter is used to determine the automatically unlock period to unlock the logic function gate.	00:00:05…**00:00:30**...18:12:15
Feedback of logic function lock status	This parameter is used to enable or disable the feedback of the logic lock status object.	**No** Yes
After Bus Voltage Recovery	This parameter is used to determine the logic function gate lock status after the bus voltage recovery.	**Lock Passive** Lock Active Lock Previous

4.12. Additional Functions – Converters

This section describes the converter function modules of the iX3. There are 2 type of converter function:

- Gate forwarding
- Format converter

4.12.1. Converters – Gate Forwarding / Format Converter

Gate forwarding is that if the input value that is specified in parameter receive, send the value that is specified in parameter to bus. The users can configure the input and output's values and data types.

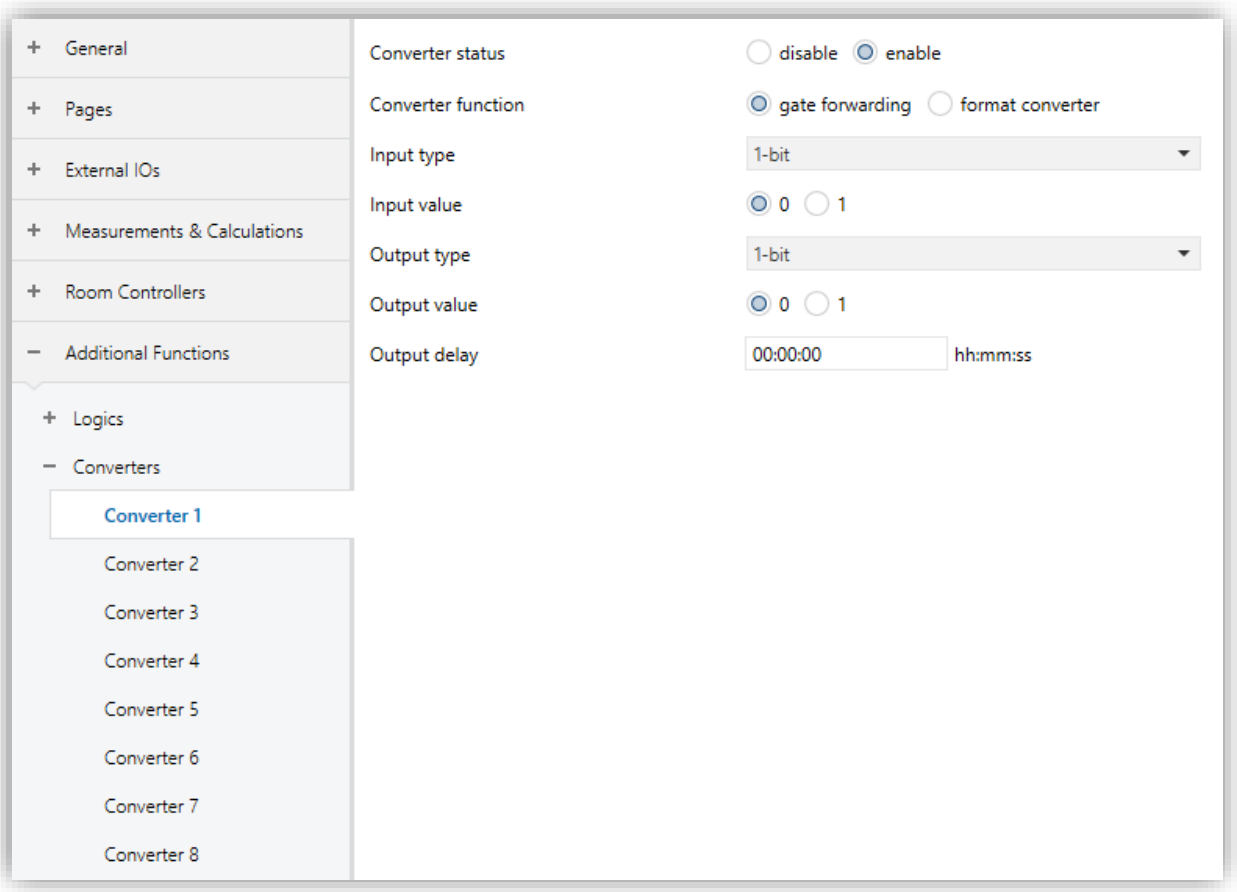


Fig. 81: Logics – Gate Forwarding Configuration

Format converter, converts the value from a data type to another data type. For example; 1-bit input object to 1-byte output object.

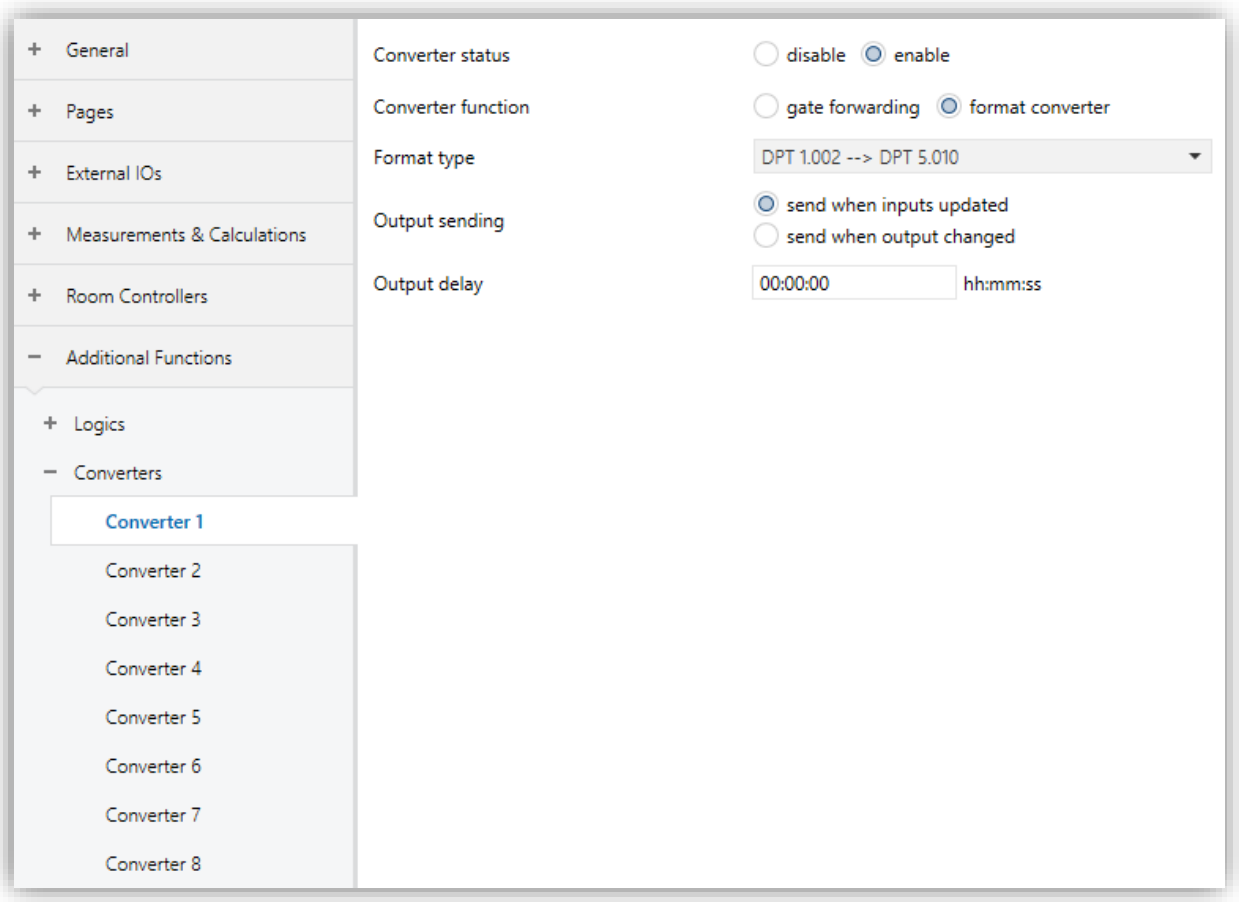


Fig. 82: Logics – Format Converter Configuration

4.12.1.1. Parameters List

PARAMETERS	DESCRIPTION	VALUES
Converter status	This parameter is used to enable or disable the converter features.	**Disable** Enable
Converter function	This parameter determines the function type of converter module.	**Gate Forwarding** Format Converter
-> Input type[1]	This parameter determines the data type of input object that must be received to output to the bus.	**1-bit** 2-bits 1-byte 2-bytes 1-byte logic 2-bytes logic 1-byte threshold 2-bytes threshold
-> Input value[1]	This parameter determines the input value that must be received to output to the bus.	Values depend on DPT selection.
-> Calculation value[1,2]	This parameter determines the value that will do the arithmetic operation with the input value.	Values depend on DPT selection.
-> Output type[1]	This parameter determines the data type of output object.	**1-bit** 2-bits 1-byte 2-bytes
-> Output value[1]	This parameter determines the value of output to be sent.	Values depend on DPT selection.
-> Output behaviour[1,3]	This parameter determines the logic value of 0 and 1.	**0: false / 1: true** 1: false / 0: true
-> Format type[4]	This parameter determines which data type will be converted to another data type.	**DPT 1.002 → DPT 5.010** 8*DPT 1.002 → DPT 5.010 DPT 5.010 → 8*DPT 1.002

		DPT 5.010 → DPT 7.001 DPT 232.600(RGB) → 3*DPT 5.010 3*DPT 5.010 → DPT 232.600(RGB) DPT 251.600(RGBW) → 4*DPT 5.010 4*DPT 5.010 → DPT 251.600(RGBW)
-> Output sending[4]	This parameter determines when the output value is sent.	**Send when inputs updated** Send when output changed
-> Output delay	This parameter determines the delay time of sending the output value. The output value is sent after the time in this parameter.	**00:00:00**….18:12:15

[1] This parameter is visible when the function "Converted function" is set to "Gate forwarding".

[2] This parameter is visible when the function "Calculation type" is **not** set to "Disabled".

[3] This parameter is visible when the function "Input type" is set to "1-Byte logic", "2-Byte logic", "1-Byte threshold", "2-Byte threshold".

[4] This parameter is visible when the function "Converted function" is set to "Format converter".

4.13. User Interface

Icon	Meaning
1	Back to the previous page.
2	Control the on/off state of "Switch" object.
3	Control the setpoint temperature by clicking or dragging.
4	Decrease the setpoint temperature.
5	Increase the setpoint temperature.
6	Open the pop-up screen to change HVAC mode.
7	Open the pop-up screen to change heating/cooling mode and manual/auto controlling.
8	Open the pop-up screen to change fan level and auto/manual controlling.

Icon	Meaning
1	Close the pop-up screen without saving of the changes.
2	Close the pop-up screen with saving of the changes.
3	Change HVAC mode between Comfort, Standby, Economy and Protection.
4	Change HVAC mode as AUTO. If this mode is selected 5, 6, 7 and 8.icons can't be touched.
5	Change HVAC mode as COMFORT.
6	Change HVAC mode as STANDBY.
7	Change HVAC mode as ECONOMY.
8	Change HVAC mode as PROTECTION.

Icon	Meaning
1	Close the pop-up screen without saving of the changes.
2	Close the pop-up screen with saving of the changes.
3	Change heating/cooling mode changing as MANUAL.
4	Change heating/cooling mode changing as AUTO.
5	Change the control mode as HEATING.
6	Change the control mode as COOLING.

Icon	Meaning
1	Close the pop-up screen without saving of the changes.
2	Close the pop-up screen with saving of the changes.
3	Change the fan speed control as MANUAL.
4	Change the fan speed control as AUTO.
5	Decrease the fan speed.
6	Increase the fan speed.
7	Change the fan speed by clicking the boxes.

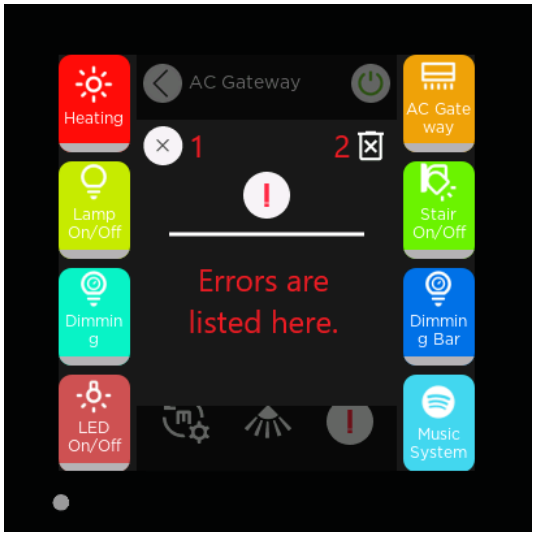
Icon	Meaning
1	Back to the previous page.
2	Control the on/off state of "Switch" object.
3	Control the setpoint temperature by clicking or dragging.
4	Decrease the setpoint temperature.
5	Increase the setpoint temperature.
6	Open the pop-up screen to change AC mode.
7	Open the pop-up screen to change AC vertical swing level and direction.
8	Open the pop-up screen to change AC fan speed.
9	Open the pop-up screen to change AC extension mode.
10	Open the pop-up screen to change AC horizontal swing level and direction.
11	Open the pop-up screen to list AC errors. Last 8 errors can be listed.

Icon	Meaning
1	Close the pop-up screen without saving of the changes.
2	Close the pop-up screen with saving of the changes.
3	Change the AC mode as HEATING.
4	Change the AC mode as COOLING.
5	Change the AC mode as DRY/DEHUMIDIFY.
6	Change the AC mode as ONLY FAN.
7	Change the AC mode as AUTO.

Icon	Meaning
1	Close the pop-up screen without saving of the changes.
2	Close the pop-up screen with saving of the changes.
3	Change the AC horizontal swing mode as FIXED.
4	Change the AC horizontal swing mode as MOVING.
5	Change the AC horizontal swing level. In fixed mode, swing level is changed by clicked the button. In moving mode, min/max level is changed by dragging the button.

Icon	Meaning
1	Close the pop-up screen without saving of the changes.
2	Close the pop-up screen with saving of the changes.
3	Change the AC vertical swing mode as FIXED.
4	Change the AC vertical swing mode as MOVING.
5	Change the AC vertical swing level. In fixed mode, swing level is changed by clicked the button. In moving mode, min/max level is changed by dragging the button.

Icon	Meaning
1	Close the pop-up screen without saving of the changes.
2	Close the pop-up screen with saving of the changes.
3	Change the AC fan speed control as MANUAL.
4	Change the AC fan speed control as AUTO by sending level 0.
5	Decrease the fan speed.
6	Increase the fan speed.
7	Change the fan speed by clicking the boxes. If fan speed control mode is AUTO, the buttons can't be touched.



Icon	Meaning
1	Close the pop-up screen without saving of the changes.
2	Clear the listed errors.

Icon	Meaning
1	Back to the previous page.
2	Control the on/off state of colour.
3	Control the colour dimming .
4	Control the colour white.
5	Control the colour temperature.
6	Control the colour by dragging on the HSV palatte.
7	Feedback of the set colour.

Icon	Meaning
1	Back to the previous page.
2	Control the on/off state of music system.
3	Set volume down.
4	Switch to previous song.
5	Control to play or stop the song.
6	Switch to next song.
7	Set volume up.

Icon	Meaning
1	Change the playing mode as REPEAT- RANDOM-LOOP- SEQUENTIAL .
2	Change the source between USB- SD-AUX-FM-BT.

Icon	Meaning
1	Mute/unmute the sound.
2	Auto/manual brightness adjustment.
3	Enter the cleaning mode.
4	Lock screen.
5	Go into settings page.
6	Change the screen brightness by dragging or clicking the slider. When clicked the slider, brightness adjustment mode is switch to MANUAL.

5. ETS Objects List & Descriptions

The iX3 KNX Room Controllers can communicate via the KNX bus line. In this section, the group objects of the iX3 KNX Room Controllers are described, which of these group objects are visible and capable of being linked with group addresses are explained in sub-sections.

No	Name	Function	DTP Type	Length	Flags				
					C	R	W	T	U
1	General	In operation	1.002	1 bit	X			X	
2	General	Navigation Led	1.001	1 bit	X		X		
3	General	Error Identification	16.000	14 bytes	X			X	
4	General	Brightness	5.001	1 byte	X		X		
5	General	Date Time	19.001	8 bytes	X		X		
6, 7, 8, 9	General	Password X Activated	1.001	1 bit	X			X	
			5.001	1 byte	X			X	
			5.004	1 byte	X			X	
			17.001	1 byte	X			X	
10, 18, 26, …, 354	Page Z Button X, Y	Disable	1.003	1 bit	X		X		
11, 19, 27, …, 355	Page Z Button X, Y	Status	1.003	1 bit	X	X		X	
12, 20, 28, …, 356	Page Z Button X, Y	Switch	1.001	1 bit	X	X	X	X	X
		Shutter UP/DOWN	1.008	1 bit	X		X	X	
		Forced Operation – Switch	2.001	2 bits	X			X	
		Forced Operation – Percent	5.001	1 byte	X			X	
		Forced Operation – Decimal	5.005	1 byte	X			X	
		Forced Operation – Scene	17.001	1 byte	X			X	
		Forced Operation – Colour	7.600	2 bytes	X			X	
		Forced Operation – Temperature	9.001	2 bytes	X			X	
		Forced Operation – Brightness	9.004	2 bytes	X			X	
		Forced Operation – RGB	232.600	3 bytes	X			X	
		Scene	18.001	1 byte	X			X	
		Mode Selection	20.102	1 byte	X		X	X	
		Sequence	1.001	1 bit	X	X		X	
			5.010	1 byte	X	X		X	
			5.001	1 byte	X	X		X	
			20.102	1 byte	X	X		X	
		Sequence A	1.001	1 bit	X	X		X	
		Sequence A (0…255)	5.010	1 byte	X	X		X	
		Sequence A (0…100%)	5.001	1 byte	X	X		X	

		Sequence A HVAC	20.102	1 byte	X	X		X	
		Counter value	5.010	1 byte	X	X		X	
			7.001	2 bytes	X	X		X	
			12.001	4 bytes	X	X		X	
		RGB Colour	232.600	3 bytes	X	X	X	X	X
		RGB – Red Colour	5.010	1 byte	X	X	X	X	X
		RGBW Colour	251.600	6 bytes	X	X	X	X	X
		RGBW – Red Colour	5.010	1 byte	X	X	X	X	X
		Thermostat Enable/Disable – A	1.003	1 bit	X	X		X	
		Thermostat Heat Cool Switch – A	1.100	1 bit	X	X		X	
		Thermostat HVAC Mode Switch – A	20.102	1 byte	X	X		X	
		Thermostat Setpoint – A	9.001	2 bytes	X	X		X	
		Thermostat Fan Level – A	5.100	1 byte	X	X		X	
		Thermostat Fan Mode – A	1.003	1 bit	X	X		X	
		Power On/Off	1.001	1 bit	X	X	X	X	X
		Song Play/Pause	1.010	1 bit	X	X	X	X	X
		Song Next/Previous	1.007	1 bit	X	X		X	
		Volume Up/Down	1.007	1 bit	X	X		X	
		Play Mode	5.010	1 byte	X	X	X	X	X
		Music Source	5.010	1 byte	x	X	X	X	X
13, 21, 29, …, 357	Page Z Button X, Y	Shutter Percentage	5.001	1 byte	X	X	X	X	X
		RGB – Green Colour	5.010	1 byte	X	X	X	X	X
		RGBW – Green Colour	5.010	1 byte	X	X	X	X	X
		Thermostat Status Fb – A	1.003	1 bit	X		X		X
		Thermostat Heat Cool Fb – A	1.100	1 bit	X		X		X
		Thermostat HVAC Mode Fb – A	20.102	1 byte	X		X		X
		Thermostat Setpoint Fb – A	9.001	2 bytes	X		X		X
		Thermostat Fan Level Fb – A	5.100	1 byte	X		X		X
		Thermostat Fan Mode Fb – A	1.003	1 bit	X		X		X
14, 22, 30, …, 358	Page Z Button X, Y	Switch - long	1.001	1 bit	X	X	X	X	X
		Dimming	3.007	4 bits	X			X	
		STOP / Lamella Adjustment	1.007	1 bit	X		X	X	
		Forced Operation – Switch	2.001	2 bits	X			X	
		Forced Operation – Percent	5.001	1 byte	X			X	
		Forced Operation – Decimal	5.005	1 byte	X			X	
		Forced Operation – Scene	17.001	1 byte	X			X	
		Forced Operation – Colour	7.600	2 bytes	X			X	

		Forced Operation – Temperature	9.001	2 bytes	X			X	
		Forced Operation – Brightness	9.004	2 bytes	X			X	
		Forced operation – RGB	232.600	3 bytes	X			X	
		Scene Store	1.003	1 bit	X	X	X		
		HVAC-Mode State	20.102	1 byte	X		X	X	X
		Sequence B	1.001	1 bit	X	X		X	
		Sequence B (0…255)	5.010	1 byte	X	X		X	
		Sequence B (0…100%)	5.001	1 byte	X	X		X	
		Sequence B HVAC	20.102	1 byte	X	X		X	
		Reset Counter	1.001	1 bit	X		X		
		RGB – Blue Colour	5.010	1 byte	X	X	X	X	X
		RGBW – Blue Colour	5.010	1 byte	X	X	X	X	X
		Thermostat Enable/Disable – B	1.003	1 bit	X	X		X	
		Thermostat Heat Cool Switch – B	1.100	1 bit	X	X		X	
		Thermostat HVAC Mode Switch – B	20.102	1 byte	X	X		X	
		Thermostat Setpoint – B	9.001	2 bytes	X	X		X	
		Thermostat Fan Level – B	5.100	1 byte	X	X		X	
		Thermostat Fan Mode – B	1.003	1 bit	X	X		X	
15, 23, 31, …, 359	Page Z Button X, Y	Dimming Absolute	5.001	1 byte	X	X	X	X	X
		RGBW – White Colour	5.010	1 byte	X	X	X	X	X
		Thermostat Status Fb – B	1.003	1 bit	X		X		X
		Thermostat Heat Cool Fb – B	1.100	1 bit	X		X		X
		Thermostat HVAC Mode Fb – B	20.102	1 byte	X		X		X
		Thermostat Setpoint Fb – B	9.001	2 bytes	X		X		X
		Thermostat Fan Level Fb – B	5.100	1 byte	X		X		X
		Thermostat Fan Mode Fb – B	1.003	1 bit	X		X		X
16, 24, 32, …, 360	Page Z Button X, Y	Upper Limit Position	1.002	1 bit	X		X		
		Sequence C	1.001	1 bit	X	X		X	
		Sequence C (0…255)	5.010	1 byte	X	X		X	
		Sequence C (0…100%)	5.001	1 byte	X	X		X	
		Sequence C HVAC	20.102	1 byte	X	X		X	
		Overflow	1.001	1 bit	X			X	
			5.010	1 byte	X			X	
		Colour Temperature	7.600	2 bytes	X	X	X	X	X
		Sequence - E	5.010	1 byte	X	X	X	X	X
17, 25, 33, …,		Lower Limit Position	1.002	1 bit	X		X		

361	Page Z Button X, Y	Sequence D	1.001	1 bit	X	X		X	
		Sequence D (0…255)	5.010	1 byte	X	X		X	
		Sequence D (0…100%)	5.001	1 byte	X	X		X	
		Sequence D HVAC	20.102	1 byte	X	X		X	
		Music Source	5.010	1 byte	X	X	X	X	X
362, 370	Input X	Disable	1.003	1 bit	X		X		
363, 371	Input X	Status	1.001	1 bit	X	X		X	
364, 372	Input X	Switch	1.001	1 bit	X	X	X	X	X
		Shutter UP/DOWN	1.008	1 bit	X		X	X	
		Forced Operation – Switch	2.001	2 bits	X			X	
		Forced Operation – Percent	5.001	1 byte	X			X	
		Forced Operation – Decimal	5.005	1 byte	X			X	
		Forced Operation – Scene	17.001	1 byte	X			X	
		Forced Operation – Colour	7.600	2 bytes	X			X	
		Forced Operation – Temperature	9.001	2 bytes	X			X	
		Forced Operation – Brightness	9.004	2 bytes	X			X	
		Forced Operation – RGB	232.600	3 bytes	X			X	
		Scene	18.001	1 byte	X			X	
		Mode selection	20.102	1 byte	X		X	X	
		Sequence	1.001	1 bit	X	X		X	
			5.010	1 byte	X	X		X	
			5.001	1 byte	X	X		X	
			20.102	1 byte	X	X		X	
		Sequence A	1.001	1 bit	X	X		X	
		Sequence A (0…255)	5.010	1 byte	X	X		X	
		Sequence A (0…100%)	5.001	1 byte	X	X		X	
		Sequence A HVAC	20.102	1 byte	X	X		X	
		Counter value	5.010	1 byte	X	X		X	
			7.001	2 bytes	X	X		X	
			12.001	4 bytes	X	X		X	
		RGB Colour	232.600	3 bytes	X	X	X	X	X
		RGB – Red Colour	5.010	1 byte	X	X	X	X	X
		RGBW Colour	251.600	6 bytes	X	X	X	X	X
		RGBW – Red Colour	5.010	1 byte	X	X	X	X	X
		Thermostat Enable/Disable – A	1.003	1 bit	X	X		X	
		Thermostat Heat Cool Switch – A	1.100	1 bit	X	X		X	
		Thermostat HVAC Mode Switch – A	20.102	1 byte	X	X		X	

		Thermostat Setpoint – A	9.001	2 bytes	X	X		X	
			9.002	2 bytes	X	X		X	
		Thermostat Fan Level – A	5.100	1 byte	X	X		X	
		Thermostat Fan Mode – A	1.003	1 bit	X	X		X	
373	Input X	RGB – Green Colour	5.010	1 byte	X	X	X	X	X
		RGBW – Green Colour	5.010	1 byte	X	X	X	X	X
		Thermostat Status Fb – A	1.003	1 bit	X		X		X
		Thermostat Heat Cool Fb – A	1.100	1 bit	X		X		X
		Thermostat HVAC Mode Fb – A	20.102	1 byte	X		X		X
		Thermostat Setpoint Fb – A	9.001	2 bytes	X		X		X
		Thermostat Fan Level Fb – A	5.100	1 byte	X		X		X
		Thermostat Fan Mode Fb – A	1.003	1 bit	X		X		X
366, 374	Input X	Switch – Long	1.001	1 bit	X	X	X	X	X
		Dimming	3.007	4 bits	X			X	
		STOP / Lamella Adjustment	1.007	1 bit	X		X	X	
		Forced operation – Switch	2.001	2 bits	X			X	
		Forced operation – Percent	5.001	1 byte	X			X	
		Forced operation – Decimal	5.005	1 byte	X			X	
		Forced operation – Scene	17.001	1 byte	X			X	
		Forced operation – Colour	7.600	2 bytes	X			X	
		Forced operation – Temperature	9.001	2 bytes	X			X	
		Forced operation – Brightness	9.004	2 bytes	X			X	
		Forced operation – RGB	232.600	3 bytes	X			X	
		Scene Store	1.003	1 bit	X	X	X		
		HVAC-Mode State	20.102	1 byte	X		X	X	X
		Sequence B	1.001	1 bit	X	X		X	
		Sequence B (0…255)	5.010	1 byte	X	X		X	
		Sequence B (0…100%)	5.001	1 byte	X	X		X	
		Sequence B HVAC	20.102	1 byte	X	X		X	
		Reset counter	1.001	1 bit	X		X		
		RGB – Blue Colour	5.010	1 byte	X	X	X	X	X
		RGBW – Blue Colour	5.010	1 byte	X	X	X	X	X
		Thermostat Enable/Disable – B	1.003	1 bit	X	X		X	
		Thermostat Heat Cool Switch – B	1.100	1 bit	X	X		X	
		Thermostat HVAC Mode Switch – B	20.102	1 byte	X	X		X	
		Thermostat Setpoint – B	9.001	2 bytes	X	X		X	
			9.002	2 bytes	X	X		X	

		Thermostat Fan Level – B	5.100	1 byte	X	X		X	
					X	X	X	X	
		Thermostat Fan Mode – B	1.003	1 bit	X	X		X	
					X	X	X	X	
367, 375	Input X	RGBW – White	5.010	1 byte	X	X	X	X	X
		Thermostat Status Fb – B	1.003	1 bit	X		X		X
		Thermostat Heat Cool Fb – B	1.100	1 bit	X		X		X
		Thermostat HVAC Mode Fb – B	20.102	1 byte	X		X		X
		Thermostat Setpoint Fb – B	9.001	2 bytes	X		X		X
		Thermostat Fan Level Fb – B	5.100	1 byte	X		X		X
		Thermostat Fan Mode Fb – B	1.003	1 bit	X		X		X
368, 376	Input X	Upper limit position	1.002	1 bit	X		X		
		Sequence C	1.001	1 bit	X	X		X	
		Sequence C (0…255)	5.010	1 byte	X	X		X	
		Sequence C (0…100%)	5.001	1 byte	X	X		X	
		Sequence C HVAC	20.102	1 byte	X	X		X	
		Overflow	1.001	1 bit	X			X	
			5.010	1 byte	X			X	
369, 377	Input X	Lower limit position	1.002	1 bit	X		X		
		Sequence D	1.001	1 bit	X	X		X	
		Sequence D (0…255)	5.010	1 byte	X	X		X	
		Sequence D (0…100%)	5.001	1 byte	X	X		X	
		Sequence D HVAC	20.102	1 byte	X	X		X	
378	Measurement Temperature Internal	Disable	1.003	1 bit	X		X		
379	Measurement Temperature Internal	Status	1.003	1 bit	X	X		X	
380	Measurement Temperature Internal	Temperature Value	9.001	2 bytes	X	X		X	
381	Measurement Temperature Internal	Temperature Calibration	9.001	2 bytes	X		X		
382	Measurement Temperature Internal	Alarm - Fault	1.005	1 bit	X			X	
383	Measurement Temperature Internal	Alarm – Low	1.005	1 bit	X			X	

384	Measurement Temperature Internal	Alarm – High	1.005	1 bit	X			X	
385	Measurement Temperature Internal	Additional Value - Bit	1.001	1 bit	X			X	
		Additional Value - Byte	5.010	1 byte	X			X	
		Additional Value - Scene	17.001	1 byte	X			X	
		Additional Value - Percentage	5.001	1 bit	X			X	
386	Measurement Humidity Internal	Disable	1.003	1 bit	X		X		
387	Measurement Humidity Internal	Status	1.003	1 bit	X	X		X	
388	Measurement Humidity Internal	Humidity Value	9.007	2 bytes	X	X		X	
389	Measurement Humidity Internal	Humidity Calibration	9.007	2 bytes	X		X		
390	Measurement Humidity Internal	Alarm - Fault	1.005	1 bit	X			X	
391	Measurement Humidity Internal	Alarm - Low	1.005	1 bit	X			X	
392	Measurement Humidity Internal	Alarm - High	1.005	1 bit	X			X	
393	Measurement Humidity Internal	Additional Value - Bit	1.001	1 bit	X			X	
		Additional Value - Byte	5.010	1 byte	X			X	
		Additional Value - Scene	17.001	1 byte	X			X	
		Additional Value - Percentage	5.001	1 bit	X			X	
394	Measurement Air Quality Internal	Disable	1.003	1 bit	X		X		
395	Measurement Air Quality Internal	Status	1.003	1 bit	X	X		X	
396	Measurement Air Quality Internal	Air Quality Value	9.008	2 bytes	X	X		X	
397	Measurement Air Quality Internal	Air Quality Calibration	9.008	2 bytes	X		X		
398	Measurement Air Quality Internal	Alarm - Fault	1.005	1 bit	X			X	

399	Measurement Air Quality Internal	Alarm - Low	1.005	1 bit	X			X	
400	Measurement Air Quality Internal	Alarm - High	1.005	1 bit	X			X	
401	Measurement Air Quality Internal	Additional Value - Bit	1.001	1 bit	X			X	
		Additional Value - Byte	5.010	1 byte	X			X	
		Additional Value - Scene	17.001	1 byte	X			X	
		Additional Value - Percentage	5.001	1 bit	X			X	
402	Measurement Brightness Internal	Disable	1.003	1 bit	X		X		
403	Measurement Brightness Internal	Status	1.003	1 bit	X	X		X	
404	Measurement Brightness Internal	Brightness Value	9.004	2 bytes	X	X		X	
405	Measurement Brightness Internal	Brightness Calibration	9.004	2 bytes	X		X		
406	Measurement Brightness Internal	Alarm - Fault	1.005	1 bit	X			X	
407	Measurement Brightness Internal	Alarm - Low	1.005	1 bit	X			X	
408	Measurement Brightness Internal	Alarm - High	1.005	1 bit	X			X	
1097	Measurement Brightness Internal	Additional Value - Bit	1.001	1 bit	X			X	
		Additional Value - Byte	5.010	1 byte	X			X	
		Additional Value - Scene	17.001	1 byte	X			X	
		Additional Value - Percentage	5.001	1 bit	X			X	
410, 418	Measurement External X	Disable	1.003	1 bit	X		X		
411, 419	Measurement External X	Status	1.003	1 bit	X	X		X	
412, 420	Measurement External X	Temperature Value	9.001	2 bytes	X	X		X	
413, 421	Measurement External X	Temperature Calibration	9.001	2 bytes	X		X		
414, 422	Measurement External X	Alarm - Fault	1.005	1 bit	X			X	

415, 423	Measurement External X	Alarm - Low	1.005	1 bit	X			X	
416, 424	Measurement External X	Alarm - High	1.005	1 bit	X			X	
417, 425	Measurement External X	Additional Value - Bit	1.001	1 bit	X			X	
		Additional Value - Byte	5.010	1 byte	X			X	
		Additional Value - Scene	17.001	1 byte	X			X	
		Additional Value - Percentage	5.001	1 bit	X			X	
426, 434 442, 450 458, 466	Calculation X	Disable	1.003	1 bit	X		X		
427, 435 443, 451 459, 467	Calculation X	Status	1.003	1 bit	X	X		X	
428, 436 444, 452 460, 468	Calculation X	Probe Input Temperature	9.001	2 bytes	X		X		
		Probe Input Humidity	9.007	2 bytes	X		X		
		Probe Input Brightness	9.004	2 bytes	X		X		
		Probe Input Proximity	7.011	2 bytes	X		X		
		Probe Input Air Quality	9.008	2 bytes	X		X		
		Probe Input Air Pressure	9.006	2 bytes	X		X		
		Probe Input Wind Speed	9.005	2 bytes	X		X		
429, 437 445, 453 461, 469	Calculation X	Probe Surveillance	1.018	1 bit	X	X		X	
430, 438 446, 454 462, 470	Calculation X	Output Temperature	9.001	2 bytes	X	X		X	
		Output Humidity	9.007	2 bytes	X	X		X	
		Output Brightness	9.004	2 bytes	X	X		X	
		Output Proximity	7.011	2 bytes	X	X		X	
		Output Air Quality	9.008	2 bytes	X	X		X	
		Output Air Pressure	9.006	2 bytes	X	X		X	
		Output Wind Speed	9.005	2 bytes	X	X		X	
431, 439 447, 455 463, 471	Calculation X	Alarm - Low	1.005	1 bit	X	X		X	
432, 440 448, 456 464, 472	Calculation X	Alarm - High	1.005	1 bit	X	X		X	
474, 545	Thermostat X	Disabling	1.003	1 bit	X		X		
		Disabling	1.003	1 bit	X	X		X	
475, 546	Thermostat X	Status	1.003	1 bit	X	X		X	
		Status	1.003	1 bit	X		X		
476, 547	Thermostat X	Switch	1.001	1 bit	X	X	X	X	X

478, 549	Thermostat X	Operation Mode	20.102	1 byte	X		X		
		Operation Mode	20.102	1 byte	X	X		X	
479, 550	Thermostat X	Operation Mode Forced	20.102	1 byte	X		X		
480, 551	Thermostat X	Operation Mode Status	20.102	1 byte	X	X		X	
		Operation Mode Feedback	20.102	1 byte	X		X		
481, 552	Thermostat X	Operation Mode [Comfort]	1.001	1 bit	X		X		
482, 553	Thermostat X	Operation Mode [Standby]	1.001	1 bit	X		X		
483, 554	Thermostat X	Operation Mode [Economy]	1.001	1 bit	X		X		
484, 555	Thermostat X	Operation Mode [Protection]	1.001	1 bit	X		X		
485, 556	Thermostat X	Heating/Cooling Switchover	1.100	1 bit	X		X		
		Heating/Cooling Switchover	1.100	1 bit	X	X		X	
486, 557	Thermostat X	Heating/Cooling Status	1.100	1 bit	X	X		X	
		Heating/Cooling Feedback	1.100	1 bit	X		X		
487, 558	Thermostat X	Heating Control Disabling	1.003	1 bit	X		X		
488, 559	Thermostat X	Heating Control Running	1.002	1 bit	X	X		X	
		Heating Control Running	1.002	1 bit	X		X		
489, 560	Thermostat X	Heating Value (1-bit)	1.001	1 bit	X	X		X	
		Heating Value (1-byte)	5.004	1 byte	X	X		X	
		Heating/Cooling Value (1-bit)	1.001	1 bit	X	X		X	
		Heating/Cooling Value (1-byte)	5.004	1 byte	X	X		X	
490, 561	Thermostat X	Heating Value Request	1.016	1 bit	X		X		
		Thermostat Heating/Cooling Value Request	1.016	1 bit			X	X	
491, 562	Thermostat X	Cooling Control Disabling	1.003	1 bit	X		X		
492, 563	Thermostat X	Cooling Control Running	1.002	1 bit	X	X		X	
		Cooling Control Running	1.002	1 bit	X		X		
493, 564	Thermostat X	Cooling Value (1-bit)	1.001	1 bit	X	X		X	
		Cooling Value (1-byte)	5.004	1 byte	X	X		X	
494, 565	Thermostat X	Cooling Value Request	1.016	1 bit	X		X		
495, 566	Thermostat X	Additional Heating Control Disabling	1.003	1 bit	X		X		
496, 567	Thermostat X	Additional Heating Control Running	1.002	1 bit	X	X		X	
497, 568	Thermostat X	Additional Heating Value(1-Bit)	1.001	1 bit	X	X		X	
		Additional Heating Value(1-Byte)	5.004	1 byte	X	X		X	
498, 569	Thermostat X	Additional Heating Value Request	1.016	1 bit	X	X		X	
499, 570	Thermostat X	Additional Cooling Control Disabling	1.003	1 bit	X		X		

500, 571	Thermostat X	Additional Cooling Control Running	1.002	1 bit	X	X		X	
501, 572	Thermostat X	Additional Cooling Value (1-Bit)	1.001	1 bit	X			X	
		Additional Cooling Value (1-Byte)	5.004	1 byte	X			X	
502, 573	Thermostat X	Additional Cooling Value Request	1.017	1 bit	X		X		
503, 574	Thermostat X	Room Temperature Output - Celsius	9.001	2 bytes	X	X		X	
		Room Temperature Input - Celsius	9.001	2 bytes	X		X		
		Room Temperature Output - Fahrenheit	9.027	2 bytes	X	X		X	
		Room Temperature Input - Fahrenheit	9.027	2 bytes	X		X		
504, 575	Thermostat X	Actual Setpoint Output	9.001	2 bytes	X	X		X	
			9.002	2 bytes	X	X		X	
			9.027	2 bytes	X	X		X	
505, 576	Thermostat X	Manual Setpoint Input	9.001	2 bytes	X		X		
			9.002	2 bytes	X		X		
			9.027	2 bytes	X		X		
506, 577	Thermostat X	Manual Setpoint Reset	1.015	1 bit	X		X		
507, 578	Thermostat X	Heating Comfort Setpoint Temperature	9.001	2 bytes	X		X		
508, 579	Thermostat X	Heating Standby Setpoint Temperature	9.001	2 bytes	X		X		
509, 580	Thermostat X	Heating Economy Setpoint Temperature	9.001	2 bytes	X		X		
510, 581	Thermostat X	Heating Protection Setpoint Temperature	9.001	2 bytes	X		X		
511, 582	Thermostat X	Cooling Comfort Setpoint Temperature	9.001	2 bytes	X		X		
512, 583	Thermostat X	Cooling Standby Setpoint Temperature	9.001	2 bytes	X		X		
513, 584	Thermostat X	Cooling Economy Setpoint Temperature	9.001	2 bytes	X		X		
514, 585	Thermostat X	Cooling Protection Setpoint Temperature	9.001	2 bytes	X		X		
515, 586	Thermostat X	Fan Controller Disable	1.003	1 bit	X		X		
516, 587	Thermostat X	Fan Controller Status	1.003	1 bit	X	X		X	
517, 588	Thermostat X	Fan Controller Working Mode	1.003	1 bit	X		X		
518, 589	Thermostat X	Fan Controller Working Mode Status	1.003	1 bit	X	X		X	

519, 590	Thermostat X	Fan Controller Proportional Output	5.001	1 byte	X	X		X	
520, 591	Thermostat X	Fan Controller Manual Step	1.007	1 bit	X		X		
		Fan Controller Manual Up/Down	1.008	1 bit	X		X		
521, 592	Thermostat X	Fan Controller Manual Stage	5.100	1 byte	X		X		
				1 byte	X	X		X	
522, 593	Thermostat X	Fan Controller Speed (1 Byte)	5.010	1 byte	X	X		X	
523, 594	Thermostat X	Fan Controller Speed Feedback (1 Byte)	5.010	1 byte	X		X		X
524, 595	Thermostat X	Fan Level 1	1.001	1 bit	X	X		X	
525, 596	Thermostat X	Fan Level 2	1.001	1 bit	X	X		X	
526, 597	Thermostat X	Fan Level 3	1.001	1 bit	X	X		X	
527, 598	Thermostat X	Fan Level 4	1.001	1 bit	X	X		X	
528, 599	Thermostat X	Fan Level 5	1.001	1 bit	X	X		X	
529, 600	Thermostat X	Fan Level 1 Feedback Input	1.001	1 bit	X		X		X
530, 601	Thermostat X	Fan Level 2 Feedback Input	1.001	1 bit	X		X		X
531, 602	Thermostat X	Fan Level 3 Feedback Input	1.001	1 bit	X		X		X
532, 603	Thermostat X	Fan Level 4 Feedback Input	1.001	1 bit	X		X		X
533, 604	Thermostat X	Fan Level 5 Feedback Input	1.001	1 bit	X		X		X
534, 605	Thermostat X	Energy Saving – Window Contact 1	1.001	1 bit	X		X		
535, 606	Thermostat X	Energy Saving – Window Contact 2	1.001	1 bit	X		X		
536, 607	Thermostat X	Energy Saving – Presence Input 1	1.001	1 bit	X		X		
537, 608	Thermostat X	Energy Saving – Presence Input 2	1.001	1 bit	X		X		
538, 609	Thermostat X	Energy Saving – Card Holder 1	1.001	1 bit	X		X		
539, 610	Thermostat X	Energy Saving – Card Holder 2	1.001	1 bit	X		X		
540, 611	Thermostat X	Temperature Limit Heating Source	9.001	2 bytes	X		X		
			9.027	2 bytes	X		X		
541, 612	Thermostat X	Temperature Limit Cooling Source	9.001	2 bytes	X		X		
			9.027	2 bytes	X		X		
542, 613	Thermostat X	Temperature Limit Additional Heating Source	9.001	2 bytes	X		X		
			9.027	2 bytes	X		X		
543, 614	Thermostat X	Temperature Limit Additional Cooling Source	9.001	2 bytes	X		X		
			9.027	2 bytes	X		X		
616, 657	Air Conditioner X	Disabling	1.003	1 bit	X		X		

617, 658	Air Conditioner X	Status	1.003	1 bit	X	X		X	
618, 659	Air Conditioner X	Switch	1.001	1 bit	X	X		X	
619, 660	Air Conditioner X	Switch Feedback	1.001	1 bit	X		X		
620, 661	Air Conditioner X	Room Temperature Input	9.001	2 bytes	X		X		
621, 662	Air Conditioner X	Room Temperature Output	9.001	2 bytes	X	X		X	
622, 663	Air Conditioner X	Setpoint Temperature	9.001	2 bytes	X	X	X	X	
623, 664	Air Conditioner X	Mode	20.105	1 byte	X	X		X	
624, 665	Air Conditioner X	Mode Feedback	20.105	1 byte	X		X		
625, 666	Air Conditioner X	Mode Auto	1.003	1 bit	X	X	X	X	X
626, 667	Air Conditioner X	Mode Heat	1.003	1 bit	X	X	X	X	X
627, 668	Air Conditioner X	Mode Cool	1.003	1 bit	X	X	X	X	X
628, 669	Air Conditioner X	Mode Dry	1.003	1 bit	X	X	X	X	X
629, 670	Air Conditioner X	Mode Fan	1.003	1 bit	X	X	X	X	X
630, 671	Air Conditioner X	Extension Mode		1 byte	X	X		X	
631, 672	Air Conditioner X	Extension Mode Feedback		1 byte	X		X		
632, 633, 634 635, 636, 637 638, 673, 674 678, 679	Air Conditioner X	Extension Mode X	1.003	1 bit	X	X	X	X	X
639, 680	Air Conditioner X	Fan Level	5.100	1 byte	X	X		X	
640, 681	Air Conditioner X	Fan Level Feedback	5.100	1 byte	X		X		
641, 642, 643 644, 645, 646 682, 683, 684 685, 686, 687	Air Conditioner X	Fan Level X	1.003	1 bit	X	X	X	X	X
647, 688	Air Conditioner X	Fan Swing Vertical Moving	1.003	1 bit	X	X	X	X	X

648, 689	Air Conditioner X	Fan Swing Horizontal Moving	1.003	1 bit	X	X	X	X	X
649, 690	Air Conditioner X	Fan Swing Level Vertical	5.010	1 byte	X	X		X	
650, 691	Air Conditioner X	Fan Swing Level Horizontal	5.010	1 byte	X	X		X	
651, 692	Air Conditioner X	Fan Swing Level Vertical Feedback	5.010	1 byte	X		X		
652, 693	Air Conditioner X	Fan Swing Level Horizontal Feedback	5.010	1 byte	X		X		
653, 694	Air Conditioner X	Error 2 Byte		2 bytes	X		X		
654, 695	Air Conditioner X	Error Text	16.000	14 bytes	X		X		
698, 726, 754 782, 810, 838 866, 894	Logic X	Lock	1.001	1 bit	X		X		
699, 727, 755 783, 811, 839 867, 895	Logic X	Status	1.001	1 bit	X	X		X	
700, 728, 756 784, 812, 840 868, 896	Logic X	External Movement	1.001	1 bit	X		X		X
701, 729, 757 785, 813, 841 869, 897	Logic X	External Brightness	9.004	2 bytes	X		X		X
702, 730, 758 786, 814, 842 870, 898,	Logic X	Brightness Threshold Lower	9.004	2 bytes	X		X		
703, 731, 759 787, 815, 843 871, 899	Logic X	Brightness Threshold Upper	9.004	2 bytes	X		X		
704, 732, 760 788, 816, 844 872, 900	Logic X	External Temperature	9.001	2 bytes	X		X		X
705, 733, 761 789, 817, 845 873, 901	Logic X	Temperature Threshold Lower	9.001	2 bytes	X		X		
706, 734, 762 790, 818, 846 874, 902	Logic X	Temperature Threshold Upper	9.001	2 bytes	X		X		
707, 708, 709 735, 736, 737 763, 764, 765 791, 792, 793 819, 820, 821 847, 848, 849	Logic X	External Input 1 / 2 / 3	1.001	1 bit	X		X		X
			5.010	1 byte	X		X		X
			7.001	2 bytes	X		X		X
			9.001	2 bytes	X		X		X
			12.001	4 bytes	X		X		X

875, 876, 877 903, 904, 905									
710, 738, 766 794, 822, 850 878, 906	Logic X	Result Status	1.002	1 bit	X	X		X	
711, 714, 717 720, 723, 739 742, 745, 748 751, 767, 770 773, 776, 779 795, 798, 801 804, 807, 823 826, 829, 832 835, 851, 854 857, 860, 863 879, 882, 885 888, 891, 907 910, 913, 916 919	Logic X	Switching	1.001	1 bit	X	X		X	
		Absolute Dimming	5.001	1 byte	X	X		X	
		Shutter	1.008	1 bit	X	X		X	
		Alarm	1.005	1 bit	X	X		X	
		Sequence	1.010	1 bit	X	X		X	
		Scene	17.001	1 byte	X	X		X	
		String (14 byte)	16.000	14 bytes	X	X		X	
		Threshold	7.001	pulses	X	X		X	
712, 715, 718 721, 724, 740 743, 746, 749 752, 768, 771 774, 777, 780 796, 799, 802 805, 808, 824 827, 830, 833 836, 852, 855 858, 861, 864 880, 883, 886 889, 892, 908 911, 914, 917 920	Logic X	Delay Time on TRUE State	7.005	2 bytes	X		X		
713, 716, 719 722, 725, 741 744, 747, 750 753, 769, 772 775, 778, 797, 800, 803, 806 809, 825, 828 831, 834, 837 853, 856, 859 862, 865, 881 884, 887, 890, 893, 909, 912 915, 918, 921	Logic X	Delay Time on FALSE State	7.005	2 bytes	X		X		
922, 933, 944 955, 966, 977 988, 999,	Converter X	Disabling	1.003	1 bit	X		X		

923, 934, 945 956, 967, 978 989, 1000	Converter X	Status	1.003	1 bit	X	X		X	
924, 935, 957 968, 979, 990 1001	Converter X	Input Byte	5.010	1 byte	X		X		
		Input 2Bytes	7.001	2 bytes	X		X		
		Input Bit	1.001	1 bit	X		X		
		Input 2Bit	2.001	2 bits	X		X		
932, 943, 946 954, 965, 976 987, 998, 1009	Converter X	Output Bit	1.001	1 bit	X	X		X	
		Output 2Bit	2.001	2 bits	X	X		X	
		Output Byte	5.010	1 byte	X	X		X	
		Output 2Bytes	7.001	2 bytes	X	X		X	
924, 925, 926 927, 928, 929 930, 931, 935 936, 937, 938 939, 940, 941 942, 946, 947 948, 949, 950 951, 952, 953 957, 958, 959 960, 961, 962 963, 964, 968 969, 970, 971 972, 973, 974 975, 979, 980 981, 982, 983 984, 985, 986 990, 991, 992 993, 994, 995 996, 997, 1001 1002, 1003, 1004, 1005, 1006, 1007, 1008	Converter X	Input Bit	1.001	1 bit	X		X		
		Input Bit:0 / 1 / 2 / 3 / 4 / 5 / 6 / 7	1.002	1 bit	X		X		
		Input Byte	5.010	1 byte	X		X		
		Input RGB	232.600	3 bytes	X		X		
		Input RGBW	251.600	6 bytes	X		X		
		Input Red / Green / Blue / White	5.001	1 byte	X		X		
925, 926, 927 928, 929, 930 931, 932, 936 937, 938, 939 940, 941, 942 943, 947, 948 949, 950, 951 952, 953, 954 958, 960, 961 962, 963, 964 965, 969, 970 971, 972, 973 974, 975, 976 981, 982, 983	Converter X	Output Bit: 0 / 1 / 2 / 3 / 4 / 5 / 6 / 7	1.002	1 bit	X	X		X	
		Output Byte	5.010	1 byte	X	X		X	
		Output RGB	232.600	3 bytes	X	X		X	
		Output RGBW	251.600	6 bytes	X	X		X	

984, 985, 986 987, 991, 992 993, 994, 995 996, 997, 998 1002, 1003, 1004, 1005, 1006, 1007, 1008, 1009		Output Red / Green / Blue / White	5.001	1 byte	X	X		X	

5.1. General Objects

This section describes the "general" group objects and their properties. General group objects, as the name suggests, indicate the general characteristics of the iX3.

Object Number	Object Name	Function	Type	Flags
1	**General**	**In operation**	**1 bit**	**CT**
This object is used to monitor the presence of the device on the KNX bus line regularly. However, monitoring telegrams can be sent cyclically on the KNX bus line. DPT: 1.002 (Boolean)				
2	**General**	**Navigation Led**	**1 bit**	**CW**
The navigation LED is controlled by this object. DPT: 1.001 (switch)				
3	**General**	**Error Identification**	**14 bytes**	**CT**
This object is used to send an error message to indicate the error type if an error occurs. DPT: 16.000 (Character String (ASCII))				
4	**General**	**Brightness**	**1 byte**	**CT**
This object is used to adjust the LCD's brightness. DPT: 5.001 (percentage (0…100%))				
5	**General**	**Date Time**	**8 bytes**	**CT**
This object is used to set date and time. Date and time are used by thermostat weekly program, timer and alarm functions. Also, receiving date and time update internal RTC time in the device. DPT: 19.001 (date time)				
6, 7, 8, 9	**General**	**Password X Activated**	**1 bit / 1 byte**	**CT**
This object is visible when password function is enabled. Selected output value is sent if entered password is true. If all passwords are the same, each password's output value is sent. DPT: According to parameter selection				

5.2. Button Objects

In this section, Button objects are described in the table below. In the first column name of the object, in the second column function name, the third column data type and fourth column the objects flags, information is given.

X: 1 … 16, **Y:** 1 … 8, **Z:** 1/2

Object Number	Object Name	Function	Type	Flags
10, 18, 26, …, 354	**Page Z Button X, Y**	**Disable**	**1 bit**	**CW**
This object is used to set the iX3 button X status. "Enabled" or "Disabled" telegram is received via this object. For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the button X will continue working. DPT: 1.003 (enable)				
11, 19, 27, …, 355	**Page Z Button X, Y**	**Status**	**1 bit**	**CRT**
This object is used to watch button X status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when input X status is changed over device. DPT: 1.003 (enable)				
12, 20, 28, …, 356	**Page Z Button X, Y**	**Switch**	**1 bit**	**CRWTU**
This communication object changes in functionality depending on the selected input function. In accordance with the parameter setting, this communication object can be switched by actuation of the input to ON, OFF or TOGGLE. DPT: 1.001 (switch)				
12, 20, 28, …, 356	**Page Z Button X, Y**	**Shutter UP/Down**	**1 bit**	**CWT**
This communication object changes in functionality depending on the selected input function. This communication object sends a shutter motion telegram UP or DOWN on the bus. By receiving telegrams, the device also recognizes movement telegrams of another sensor, e.g., parallel operation. DPT: 1.008 (up/down)				

12, 20, 28, …, 356	**Page Z Button X, Y**	**Forced Operation**	**2 bit / 1 byte / 2 bytes/ 3 bytes**	**CT**

This communication object changes in functionality depending on the selected input function. This communication object sends a value on the bus with short operation when opening or closing of the contact. Depending on the configuration, the data type of this object changes. forced, percent value, decimal value, Scene number, temperature value, brightness value and percent value (RGB) can be performed on this object.
DPT: According to parameter selection

12, 20, 28, …, 356	**Page Z Button X, Y**	**Scene**	**1 byte**	**CT**

This communication object stores the value of the active scene number (1 - 64).
DPT: 18.001 (scene control)

12, 20, 28, …, 356	**Page Z Button X, Y**	**Mode Selection**	**1 byte**	**CWT**

This object keeps the active HVAC state that can be toggled through press events.
Note: There can be up to 4 different HVAC state (comfort, standby, economy, building protection) selected and each press event toggles through the HVAC states that are set as available in the parameter list.
DPT: 20.102 (HVAC mode)

12, 20, 28, …, 356	**Page Z Button X, Y**	**Sequence**	**1 bit / 1 byte**	**CRT**

This object keeps the current command that can be toggled through press events. Used for "Single Object" parameter selection.
Note: Each state (State A, B, C, D) holds a different value with adjustable data length. Each press event puts the next available state's data to the "Sequence" object.
DPT: According to parameter selection

12, 20, 28, …, 356	**Page Z Button X, Y**	**Sequence A**	**1 bit / 1 byte**	**CRT**

This object keeps the current command that can be toggled through press events. Used for "Multiple Object" parameter selection.
Note: Each object (Object A, B, C, D) holds a different value with adjustable data length. Each press event puts the next available state's data to the "Sequence X" object and whichever object is holds the current state is sent to bus with its data.
DPT: According to parameter selection

12, 20, 28, …, 356	**Page Z Button X, Y**	**Counter value**	**1 byte / 2 bytes/ 4 bytes**	**CRT**

This object keeps the current value of the press counter.
DPT: According to parameter selection

12, 20, 28, …, 356	**Page Z Button X, Y**	**RGB Red Colour / RGB Colour**	**1 byte / 3 bytes**	**CRWTU**

This object either keeps the 1-Byte Red value of the RGB, or keeps the entire 3-Byte RGB value. Decision is made in the parameter list as either "1 object of 3 bytes" or 3 objects of 1 byte".
DPT: 5.010 (counter pulses) / 232.600 (RGB value)

12, 20, 28, …, 356	**Page Z Button X, Y**	**RG BW Red Colour/ RGBW Colour**	**6 bytes/ 1 byte**	**CRWTU**

If the "object type" is set to "1 object", this object keeps the 6-Byte RGBW value, but, if the "object type" is set to "4 objects", this object keeps the 1-Byte Red value of the RGBW.
DPT: 251.600 (RGBW value) / 5.010 (counter pulses)

12, 20, 28, …, 356	**Page Z Button X, Y**	**Thermostat Enable/Disable - A**	**1 bit**	**CRT**

This object can be used via thermostat extension control function for external thermostat on short press operation. Thermostat status is controlled via this object.
DPT: 1.003 (enable)

12, 20, 28, …, 356	**Page Z Button X, Y**	**Thermostat Heat Cool Switch - A**	**1 bit**	**CRT**

This object can be used via thermostat extension control function for external thermostat on short press operation. Heating/cooling mode changeover is controlled via this object.
DPT: 1.100 (cooling/heating)

12, 20, 28, …, 356	**Page Z Button X, Y**	**Thermostat HVAC Mode Switch - A**	**1 byte**	**CRT**

This object can be used via thermostat extension control function for external thermostat. The HVAC operating mode is controlled via this object.
DPT: 20.102 (HVAC mode)

12, 20, 28, …, 356	**Page Z Button X, Y**	**Thermostat Setpoint - A**	**2 bytes**	**CRT**

This object can be used via thermostat extension control function for external thermostat on short press operation. The setpoint temperature is controlled via this object.
DPT: 9.001 (temperature °C)

12, 20, 28, …, 356	**Page Z Button X, Y**	**Thermostat Fan Level - A**	**1 byte**	**CRT**

This object can be used via thermostat extension control function for external thermostat on short press operation. The fan speed is controlled via this object.
DPT: 5.100 (switch)

12, 20, 28, …, 356	**Page Z Button X, Y**	**Thermostat Fan Mode - A**	**1 bit**	**CRT**

This object can be used via thermostat extension control function for external thermostat on short press operation. The fan auto/manual working mode is controlled via this object.
DPT: 1.003 (enable)

12, 20, 28, …, 356	**Page Z Button X, Y**	**RGB Green Colour**	**1 byte**	**CRWTU**

This object keeps the 1-Byte green value of RGB if "3 objects of 1 Byte" option is selected in the parameter list.
DPT: 5.010 (counter pulses)

13, 21, 29, …, 357	**Page Z Button X, Y**	**RGBW Green Colour**	**1 byte**	**CRWTU**

If the "object type" is set to "4 objects", this object keeps the 1-Byte Green value of the RGBW.
DPT: 5.010 (counter pulses)

13, 21, 29, …, 357	**Page Z Button X, Y**	**Thermostat Status Fb - A**	**1 bit**	**CWU**

This object can be used via thermostat extension control function for external thermostat on short press operation. The thermostat status is watched via this object.
DPT: 1.003 (enable)

13, 21, 29, …, 357	**Page Z Button X, Y**	**Thermostat Heat Cool Fb - A**	**1 bit**	**CWU**

This object can be used via thermostat extension control function for external thermostat on short press operation. The heating/cooling mode is watched via this object.
DPT: 1.100 (cooling/heating)

13, 21, 29, …, 357	**Page Z Button X, Y**	**Thermostat HVAC Fb - A**	**1 byte**	**CWU**

This object can be used via thermostat extension control function for external thermostat on short press operation. The HVAC operating mode is watched via this object.
DPT: 20.102 (HVAC mode)

13, 21, 29, …, 357	**Page Z Button X, Y**	**Thermostat Setpoint Fb - A**	**2 bytes**	**CWU**

This object can be used via thermostat extension control function for external thermostat on short press operation. The setpoint temperature is watched via this object.
DPT: 9.001 (temperature (°C))

13, 21, 29, …, 357	**Page Z Button X, Y**	**Thermostat Fan Level Fb - A**	**1 byte**	**CWU**

This object can be used via thermostat extension control function for external thermostat on short press operation. The fan speed is watched via this object.
DPT: 1.003 (enable)

13, 21, 29, …, 357	**Page Z Button X, Y**	**Thermostat Fan Mode Fb - A**	**1 bit**	**CWU**

This object can be used via thermostat extension control function for external thermostat on short press operation. The fan auto/manual working mode is watched via this object.
DPT: 1.003 (enable)

14, 22, 30, …, 358	**Page Z Button X, Y**	**Switch - Long**	**1 bit**	**CRWTU**

This communication object changes in functionality depending on the selected input function. In accordance with the parameter setting, this communication object can be switched by actuation of the input to ON, OFF or TOGGLE.
DPT: 1.001 (switch)

14, 22, 30, …, 358	**Page Z Button X, Y**	**Dimming**	**4 bits**	**CT**

This communication object changes in functionality depending on the selected input function. In accordance with the parameter setting, A long operation at the input has the effect that BRIGHTER or DARKER dim telegrams are sent via this communication object on the bus. A STOP telegram is sent and the cyclic sending of dim telegrams is stopped at the end of actuation with START-STOP-DIMMING.
DPT: 3.007 (dimming control)

14, 22, 30, …, 358	**Page Z Button X, Y**	**STOP / Lamella Adjustment**	**1 bit**	**CT**

This communication object changes in functionality depending on the selected input function. This communication object sends a STOP telegram or slat adjustment.
DPT: 1.007 (step)

14, 22, 30, …, 358	**Page Z Button X, Y**	**Forced Operation - Long**	**2 bits / 1 byte / 2 bytes/ 3 bytes**	**CT**

This communication object changes in functionality depending on the selected input function. This communication object sends a value on the bus with short operation when opening or closing of the contact. Depending on the configuration, the data type of this object changes. forced, percent value, decimal value, Scene number, temperature value, brightness value and percent value (RGB) can be performed on this object.
DPT: According to parameter selection

14, 22, 30, …, 358	**Page Z Button X, Y**	**Scene Store**	**1 bit**	**CRW**

This communication object, when active, decides whether to call or store the preset 8-bit scene number in the parameter list. When the store scene object is enabled the preset scene number is stored, but, when disabled preset scene number is called to be active.
DPT: 1.003 (enable)

14, 22, 30, …, 358	**Page Z Button X, Y**	**HVAC-Mode State**	**1 byte**	**CWTU**

This object takes the HVAC state changed via the bus.
Note: Whenever this object is updated from the bus, the HVAC state that this object holds will be considered as the valid HVAC state and press events will act as if the last HVAC state is what this object is updated with.
DPT: 20.102 (HVAC mode)

14, 22, 30, …, 358	**Page Z Button X, Y**	**Sequence B**	**1 bit / 1 byte**	**CRT**

This object keeps the current command that can be toggled through press events. Used for "Multiple Object" parameter selection.
Note: Each object (Object A, B, C, D) holds a different value with adjustable data length. Each press event puts the next available state's data to the "Sequence X" object and whichever object is holds the current state is sent to bus with its data.
DPT: According to parameter selection

14, 22, 30, …, 358	**Page Z Button X, Y**	**Reset Counter**	**1 bit**	**CW**

This object is used to reset the counter value to preset start value that can be set from parameter list.
DPT: According to parameter selection

14, 22, 30, …, 358	**Page Z Button X, Y**	**RGB Blue Colour**	**1 byte**	**CRWTU**

This object keeps the 1-Byte blue value of RGB if "3 objects of 1 Byte" option is selected in the parameter list.
DPT: 5.010 (counter pulses)

14, 22, 30, …, 358	**Page Z Button X, Y**	**RGBW Blue Colour**	**1 byte**	**CRWTU**

If the "object type" is set to "4 objects", this object keeps the 1-Byte Blue value of the RGBW.
DPT: 5.010 (counter pulses)

14, 22, 30, …, 358	**Page Z Button X, Y**	**Thermostat Enable/Disable – B**	**1 bit**	**CRT**

This object can be used via thermostat extension control function for external thermostat on long press operation. The thermostat status is controlled via this object.
DPT: 1.003 (enable)

14, 22, 30, …, 358	**Page Z Button X, Y**	**Thermostat Heat Cool Switch – B**	**1 bit**	**CRT**

This object can be used via thermostat extension control function for external thermostat on long press operation. Heating/cooling mode changeover is controlled via this object.
DPT: 1.100 (cooling/heating)

14, 22, 30, …, 358	**Page Z Button X, Y**	**Thermostat HVAC Mode Switch – B**	**1 byte**	**CRT**

This object can be used via thermostat extension control function for external thermostat on long press operation. The HVAC operating mode is controlled via this object.
DPT: 20.102 (HVAC mode)

14, 22, 30, …, 358	**Page Z Button X, Y**	**Thermostat Setpoint – B**	**2 bytes**	**CRT**

This object can be used via thermostat extension control function for external thermostat on long press operation. The setpoint temperature is controlled via this object.
DPT: 9.001 (temperature °C)

14, 22, 30, …, 358	**Page Z Button X, Y**	**Thermostat Fan Level – B**	**1 byte**	**CRT**

This object can be used via thermostat extension control function for external thermostat on long press operation. The fan speed is controlled via this object.
DPT: 5.100 (switch)

14, 22, 30, …, 358	**Page Z Button X, Y**	**Thermostat Fan Mode – B**	**1 bit**	**CRT**

This object can be used via thermostat extension control function for external thermostat on long press operation. The fan auto/manual working mode is controlled via this object.
DPT: 1.003 (enable)

15, 23, 31, …, 359	**Page Z Button X, Y**	**RGBW White Colour**	**1 byte**	**CRWTU**

If the "object type" is set to "4 objects", this object keeps the 1-Byte White value of the RGBW.
Note: White value is the colour temperature.
DPT: 5.010 (counter pulses)

15, 23, 31, …, 359	**Page Z Button X, Y**	**Thermostat Status Fb – B**	**1 bit**	**CWU**

This object can be used via thermostat extension control function for external thermostat on long press operation. The thermostat status is watched via this object.
DPT: 1.003 (enable)

15, 23, 31, …, 359	**Page Z Button X, Y**	**Thermostat Heat Cool Fb – B**	**1 bit**	**CWU**

This object can be used via thermostat extension control function for external thermostat on long press operation. The heating/cooling mode is watched via this object.
DPT: 1.100 (cooling/heating)

15, 23, 31, …, 359	**Page Z Button X, Y**	**Thermostat HVAC Fb – B**	**1 byte**	**CWU**

This object can be used via thermostat extension control function for external thermostat. The HVAC operating mode is watched via this object.
DPT: 20.102 (HVAC mode)

15, 23, 31, …, 359	**Page Z Button X, Y**	**Thermostat Setpoint Fb – B**	**2 bytes**	**CWU**

This object can be used via thermostat extension control function for external thermostat on long press operation. The setpoint temperature is watched via this object.
DPT: 9.001 (temperature (°C))

15, 23, 31, …, 359	**Page Z Button X, Y**	**Thermostat Fan Level Fb – B**	**1 byte**	**CWU**

This object can be used via thermostat extension control function for external thermostat on long press operation. The fan speed is watched via this object.
DPT: 1.003 (enable)

15, 23, 31, …, 359	**Page Z Button X, Y**	**Thermostat Fan Mode Fb - B**	**1 bit**	**CWU**

This object can be used via thermostat extension control function for external thermostat on long press operation. The fan auto/manual working mode is watched via this object.
DPT: 1.003 (enable)

16, 24, 32, …, 360	**Page Z Button X, Y**	**Upper Limit Position**	**1 bit**	**CW**

This object is used for the shutter actuator indicates if it is in the lower limit position ("shutter/blind closed"). The object is intended for a 1-button operation. '0' is no lower limit operation, '1' lower end operation.
DPT: 1.002 (boolean)

16, 24, 32, …, 360	**Page Z Button X, Y**	**Sequence C**	**1 bit / 1 byte**	**CRT**

This object keeps the current command that can be toggled through press events. Used for "Multiple Object" parameter selection.
Note: Each object (Object A, B, C, D) holds a different value with adjustable data length. Each press event puts the next available state's data to the "Sequence X" object and whichever object is holds the current state is sent to bus with its data.
DPT: According to parameter selection

16, 24, 32, …, 360	**Page Z Button X, Y**	**Overflow**	**1 bit / 1 byte**	**CRWT**

This object is sent to bus with the preset value from the parameter list when the counter value exceeds the preset end value of the counter.
DPT: 1.001 (switch) / 5.010 (counter pulses)

17, 25, 33, …, 361	**Page Z Button X, Y**	**Lower Limit Position**	**1 bit**	**CW**

This object is used for the shutter actuator indicates if it is in the lower limit position ("shutter/blind closed"). The object is intended for a 1-button operation. '0' is no lower limit operation, '1' lower end operation.
DPT: 1.002 (boolean)

17, 25, 33, …, 361	**Page Z Button X, Y**	**Sequence D**	**1 bit / 1 byte**	**CRT**

This object keeps the current command that can be toggled through press events. Used for "Multiple Object" parameter selection.
Note: Each object (Object A, B, C, D) holds a different value with adjustable data length. Each press event puts the next available state's data to the "Sequence X" object and whichever object is holds the current state is sent to bus with its data.
DPT: According to parameter selection

17, 25, 33, …, 361	**Page Z Button X, Y**	**Power On/Off**	**1 bit**	**CRWTU**

This object is used to send the music on/off controlling telegram to the bus, to control the power of the music module, and to receive feedback from the switch status of the background music on the bus. This object is visible if button function is selected as "music control".
DPT: 1.001 (switch)

17, 25, 33, …, 361	**Page Z Button X, Y**	**Song Play/Pause**	**1 bit**	**CRWTU**

This object is used to play/stop the music in the music module and can also receive status feedback. This object is visible if button function is selected as "music control".
DPT: 1.010 (start/stop)

17, 25, 33, …, 361	**Page Z Button X, Y**	**Song Next/Previous**	**1 bit**	**CRT**

This object is used to switch the playing song of the music module to the previous song / the next song. This object is visible if button function is selected as "music control".
DPT: 1.007 (step)

17, 25, 33, … , 361	**Page Z Button X, Y**	**Volume Up/Down**	**1 bit**	**CRT**

This object is used to adjust the volume of the music module.
DPT: 1.007 (step)

17, 25, 33, …, 361	**Page Z Button X, Y**	**Play Mode**	**1 byte**	**CRWTU**

This object is used to send the control telegram of the music playing mode, and the received telegram needs to be the telegram specified by the parameter to update the display status on the screen.
DPT: 5.010 (counter pulses)

17, 25, 33, …, 361	**Page Z Button X, Y**	**Music Source**	**1 byte**	**CRWTU**

This object is used for receiving the status feedback telegram of the music playing mode, and the received telegram needs to be the telegram specified by the parameter to update the display status on the screen.
DPT: 5.010 (counter pulses)

5.3. External Input Objects

In this section, Input X objects are described in the table below. In the first column name of the object, in the second column function name, the third column data type and fourth column the objects flags, information is given.

X: 1 … 2

Object Number	Object Name	Function	Type	Flags
362, 370	**Input X**	**Disable**	**1 bit**	**CW**
This object is used to set the iX3 external input X status. "Enabled" or "Disabled" telegram is received via this object. For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the external input X will continue working. DPT: 1.003 (enable)				
363, 371	**Input X**	**Status**	**1 bit**	**CRT**
This object is used to watch input X status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when input X status is changed over device. DPT: 1.001 (switch)				
364, 372	**Input X**	**Switch**	**1 bit**	**CRWTU**
This communication object changes in functionality depending on the selected input function. In accordance with the parameter setting, this communication object can be switched by actuation of the input to ON, OFF or TOGGLE. DPT: 1.001 (switch)				
364, 372	**Input X**	**Shutter UP/Down**	**1 bit**	**CWT**
This communication object changes in functionality depending on the selected input function. This communication object sends a shutter motion telegram UP or DOWN on the bus. By receiving telegrams, the device also recognizes movement telegrams of another sensor, e.g. parallel operation. DPT: 1.008 (up/down)				

364, 372	**Input X**	**Forced Operation**	**2 bit / 1 byte / 2 bytes/ 3 bytes**	**CT**

This communication object changes in functionality depending on the selected input function. This communication object sends a value on the bus with short operation when opening or closing of the contact. Depending on the configuration, the data type of this object changes. forced, percent value, decimal value, Scene number, temperature value, brightness value and percent value (RGB) can be performed on this object.
DPT: According to parameter selection

364, 372	**Input X**	**Scene**	**1 byte**	**CT**

This communication object stores the value of the active scene number (1 - 64).
DPT: 18.001 (scene control)

364, 372	**Input X**	**Mode selection**	**1 byte**	**CWT**

This object keeps the active HVAC state that can be toggled through press events.
Note: There can be up to 4 different HVAC state (comfort, standby, economy, building protection) selected and each press event toggles through the HVAC states that are set as available in the parameter list.
DPT: 20.102 (HVAC mode)

364, 372	**Input X**	**Sequence**	**1 bit / 1 byte**	**CRT**

This object keeps the current command that can be toggled through press events. Used for "Single Object" parameter selection.
Note: Each state (State A, B, C, D) holds a different value with adjustable data length. Each press event puts the next available state's data to the "Sequence" object.
DPT: According to parameter selection

364, 372	**Input X**	**Sequence A**	**1 bit / 1 byte**	**CRT**

This object keeps the current command that can be toggled through press events. Used for "Multiple Object" parameter selection.
Note: Each object (Object A, B, C, D) holds a different value with adjustable data length. Each press event puts the next available state's data to the "Sequence X" object and whichever object is holds the current state is sent to bus with its data.
DPT: According to parameter selection

364, 372	**Input X**	**Counter value**	**1 byte / 2 bytes/ 4 bytes**	**CRT**

This object keeps the current value of the press counter.
DPT: According to parameter selection

364, 372	**Input X**	**RGB Red Colour / RGB Colour**	**1 byte / 3 bytes**	**CRWTU**

This object either keeps the 1-Byte Red value of the RGB, or keeps the entire 3-Byte RGB value. Decision is made in the parameter list as either "1 object of 3 bytes" or 3 objects of 1 byte".
DPT: 5.010 (counter pulses) / 232.600 (RGB value)

364, 372	**Input X**	**RGBW Red Colour / RGBW Colour**	**6 bytes/ 1 byte**	**CRWTU**

If the "object type" is set to "1 object", this object keeps the 6-Byte RGBW value, but, if the "object type" is set to "4 objects", this object keeps the 1-Byte Red value of the RGBW.
DPT: 251.600 (RGBW value) / 5.010 (counter pulses)

364, 372	**Input X**	**Thermostat Enable/Disable - A**	**1 bit**	**CRT**

This object can be used via thermostat extension control function for external thermostat on short press operation. Thermostat status is controlled via this object.
DPT: 1.003 (enable)

364, 372	**Input X**	**Thermostat Heat Cool Switch - A**	**1 bit**	**CRT**

This object can be used via thermostat extension control function for external thermostat on short press operation. Heating/cooling mode changeover is controlled via this object.
DPT: 1.100 (cooling/heating)

364, 372	**Input X**	**Thermostat HVAC Mode Switch - A**	**1 byte**	**CRT**

This object can be used via thermostat extension control function for external thermostat on short press operation. The HVAC operating mode is watched via this object.
DPT: 20.102 (HVAC mode)

364, 372	**Input X**	**Thermostat Setpoint - A**	**2 bytes**	**CRT**

This object can be used via thermostat extension control function for external thermostat on short press operation. The setpoint temperature is controlled via this object.
DPT: 9.001 (temperature °C)

364, 372	Input X	Thermostat Fan Level - A	1 byte	CRT

This object can be used via thermostat extension control function for external thermostat on short press operation. The fan speed is controlled via this object.
DPT: 5.100 (switch)

364, 372	Input X	Thermostat Fan Mode - A	1 bit	CRT

This object can be used via thermostat extension control function for external thermostat on short press operation. The fan auto/manual working mode is controlled via this object.
DPT: 1.003 (enable)

365, 373	Input X	RGB Green Colour	1 byte	RWCTU

This object keeps the 1-Byte green value of RGB if "3 objects of 1 Byte" option is selected in the parameter list.
DPT: 5.010 (counter pulses)

365, 373	Input X	RGBW Green Colour	1 byte	RWCTU

If the "object type" is set to "4 objects", this object keeps the 1-Byte Green value of the RGBW.
DPT: 5.010 (counter pulses)

365, 373	Input X	Thermostat Status Fb - A	1 bit	CWU

This object can be used via thermostat extension control function for external thermostat on short press operation. The thermostat status is watched via this object.
DPT: 1.003 (enable)

365, 373	Input X	Thermostat Heat Cool Fb - A	1 bit	CWU

This object can be used via thermostat extension control function for external thermostat on short press operation. The heating/cooling mode is watched via this object.
DPT: 1.100 (cooling/heating)

365, 373	Input X	Thermostat HVAC Fb - A	1 byte	CWU

This object can be used via thermostat extension control function for external thermostat on short press operation. The HVAC operating mode is watched via this object.
DPT: 20.102 (HVAC mode)

365, 373	Input X	Thermostat Setpoint Fb - A	2 bytes	CWU

This object can be used via thermostat extension control function for external thermostat on short press operation. The setpoint temperature is watched via this object.
DPT: 9.001 (temperature (°C))

365, 373	Input X	Thermostat Fan Level Fb - A	1 byte	CWU

This object can be used via thermostat extension control function for external thermostat on short press operation. The fan speed is controlled via this object.
DPT: 1.003 (enable)

365, 373	Input X	Thermostat Fan Mode Fb - A	1 bit	CWU

This object can be used via thermostat extension control function for external thermostat on short press operation. The fan auto/manual working mode is watched via this object.
DPT: 1.003 (enable)

366, 374	Input X	Switch - Long	1 bit	CRWTU

This communication object changes in functionality depending on the selected input function. In accordance with the parameter setting, this communication object can be switched by actuation of the input to ON, OFF or TOGGLE.
DPT: 1.001 (switch)

366, 374	Input X	Dimming	4 bits	CT

This communication object changes in functionality depending on the selected input function. In accordance with the parameter setting, A long operation at the input has the effect that BRIGHTER or DARKER dim telegrams are sent via this communication object on the bus. A STOP telegram is sent and the cyclic sending of dim telegrams is stopped at the end of actuation with START-STOP-DIMMING.
DPT: 3.007 (dimming control)

366, 374	Input X	STOP / Lamella Adjustment	1 bit	CWT

This communication object changes in functionality depending on the selected input function. This communication object sends a STOP telegram or slat adjustment.
DPT: 1.007 (step)

366, 374	Input X	Forced Operation – Long	2 bit / 1 byte / 2 bytes/ 3 bytes	CT

This communication object changes in functionality depending on the selected input function. This communication object sends a value on the bus with short operation when opening or closing of the contact. Depending on the configuration, the data type of this object changes. forced, percent value, decimal value, Scene number, temperature value, brightness value and percent value (RGB) can be performed on this object.
DPT: According to parameter selection

366, 374	Input X	Scene Store	1 bit	CRW

This communication object, when active, decides whether to call or store the preset 8-bit scene number in the parameter list. When the store scene object is enabled the preset scene number is stored, but, when disabled preset scene number is called to be active.
DPT: 1.003 (enable)

366, 374	Input X	HVAC-Mode State	1 byte	CWTU

This object takes the HVAC state changed via the bus.
Note: Whenever this object is updated from the bus, the HVAC state that this object holds will be considered as the valid HVAC state and press events will act as if the last HVAC state is what this object is updated with.
DPT: 20.102 (HVAC mode)

366, 374	Input X	Sequence B	1 bit / 1 byte	CRT

This object keeps the current command that can be toggled through press events. Used for "Multiple Object" parameter selection.
Note: Each object (Object A, B, C, D) holds a different value with adjustable data length. Each press event puts the next available state's data to the "Sequence X" object and whichever object is holds the current state is sent to bus with its data.
DPT: According to parameter selection

366, 374	Input X	Reset counter	1 bit	CW

This object is used to reset the counter value to preset start value that can be set from parameter list.
DPT: According to parameter selection

366, 374	Input X	RGB Blue Colour	1 byte	RWCTU

This object keeps the 1-Byte green value of RGB if "3 objects of 1 Byte" option is selected in the parameter list.
DPT: 5.010 (counter pulses)

366, 374	Input X	RGBW Blue Colour	1 byte	RWCTU

If the "object type" is set to "4 objects", this object keeps the 1-Byte Green value of the RGBW.
DPT: 5.010 (counter pulses)

366, 374	Input X	Thermostat Enable/Disable – B	1 bit	CRT

This object can be used via thermostat extension control function for external thermostat on long press operation. The thermostat status is controlled via this object.
DPT: 1.003 (enable)

366, 374	Input X	Thermostat Heat Cool Switch – B	1 bit	CRT

This object can be used via thermostat extension control function for external thermostat on long press operation. Heating/cooling mode changeover is controlled via this object.
DPT: 1.100 (cooling/heating)

366, 374	Input X	Thermostat HVAC Mode Switch – B	1 byte	CRT

This object can be used via thermostat extension control function for external thermostat on long press operation. The HVAC operating mode is controlled via this object.
DPT: 20.102 (HVAC mode)

366, 374	Input X	Thermostat Setpoint – B	2 bytes	CRT

This object can be used via thermostat extension control function for external thermostat on long press operation. The setpoint temperature is controlled via this object.
DPT: 9.001 (temperature °C)

366, 374	Input X	Thermostat Fan Level – B	1 byte	CRT

This object can be used via thermostat extension control function for external thermostat on long press operation. The fan speed is controlled via this object.
DPT: 5.100 (switch)

366, 374	Input X	Thermostat Fan Mode – B	1 bit	CRT

This object can be used via thermostat extension control function for external thermostat on long press operation. The fan auto/manual working mode is controlled via this object.
DPT: 1.003 (enable)

367, 375	Input X	RGBW White Colour	1 byte	RWCTU

If the "object type" is set to "4 objects", this object keeps the 1-Byte White value of the RGBW.
Note: White value is the colour temperature.
DPT: 5.010 (counter pulses)

367, 375	Input X	Thermostat Status Fb – B	1 bit	CWU

This object can be used via thermostat extension control function for external thermostat on long press operation. The thermostat status is watched via this object.
DPT: 1.003 (enable)

367, 375	Input X	Thermostat Heat Cool Fb – B	1 bit	CWU

This object can be used via thermostat extension control function for external thermostat on long press operation. The heating/cooling mode is watched via this object.
DPT: 1.100 (cooling/heating)

367, 375	Input X	Thermostat HVAC Fb – B	1 byte	CWU

This object can be used via thermostat extension control function for external thermostat. The HVAC operating mode is watched via this object.
DPT: 20.102 (HVAC mode)

367, 375	Input X	Thermostat Setpoint Fb – B	2 bytes	CWU

This object can be used via thermostat extension control function for external thermostat on long press operation. The setpoint temperature is controlled via this object.
DPT: 9.001 (temperature (°C))

367, 375	Input X	Thermostat Fan Level Fb – B	1 byte	CWU

This object can be used via thermostat extension control function for external thermostat on long press operation. The fan speed is watched via this object.
DPT: 1.003 (enable)

367, 375	Input X	Thermostat Fan Mode Fb – B	1 bit	CWU

This object can be used via thermostat extension control function for external thermostat on long press operation. The fan auto/manual working mode is watched via this object.
DPT: 1.003 (enable)

368, 376	Input X	Upper Limit Position	1 bit	CW

This object is used for the shutter actuator indicates if it is in the lower limit position ("shutter/blind closed"). The object is intended for a 1-button operation. '0' is no lower limit operation, '1' lower end operation.
DPT: 1.002 (boolean)

368, 376	Input X	Sequence C	1 bit / 1 byte	CRT

This object keeps the current command that can be toggled through press events. Used for "Multiple Object" parameter selection.
Note: Each object (Object A, B, C, D) holds a different value with adjustable data length. Each press event puts the next available state's data to the "Sequence X" object and whichever object is holds the current state is sent to bus with its data.
DPT: According to parameter selection

368, 376	Input X	Overflow	1 bit / 1 byte	CT

This object is sent to bus with the preset value from the parameter list when the counter value exceeds the preset end value of the counter.
DPT: 1.001 (switch) / 5.010 (counter pulses)

369, 377	Input X	Lower Limit Position	1 bit	CW

This object is used for the shutter actuator indicates if it is in the lower limit position ("shutter/blind closed"). The object is intended for a 1-button operation. '0' is no lower limit operation, '1' lower end operation.
DPT: 1.002 (boolean)

369, 377	Input X	Sequence D	1 bit / 1 byte	CRT

This object keeps the current command that can be toggled through press events. Used for "Multiple Object" parameter selection.
Note: Each object (Object A, B, C, D) holds a different value with adjustable data length. Each press event puts the next available state's data to the "Sequence X" object and whichever object is holds the current state is sent to bus with its data.
DPT: According to parameter selection

5.4. Measurements Objects

In this section, Measurement's objects are described in the table below. In the first column name of the object, in the second column function name, the third column data type and fourth column the objects flags, information is given.

5.4.1. Temperature Measurement Objects

Object Number	Object Name	Function	Type	Flags
378	**Measurement Temperature Internal**	**Disable**	**1 bit**	**CW**
This object is used to set the iX3 internal temperature sensor status. "Enabled" or "Disabled" telegram is received via this object. For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the internal temperature sensor will continue the measurement. On disabled, any telegram isn't transmitted to the KNX bus. DPT: 1.003 (enable)				
379	**Measurement Temperature Internal**	**Status**	**1 bit**	**CRT**
This object is used to watch internal temperature sensor status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when internal temperature sensor status is changed over device. DPT: 1.003 (enable)				
380	**Measurement Temperature Internal**	**Temperature Value**	**2 bytes**	**CRT**
This object is used to send the measurement value that is measured by the sensor after calibrating it. Each measurement value can be calibrated via "Adjustment factor" parameter or "Calibration" object. Depending on the parameter configuration, the calculated data can be sent to the bus line periodically or according to the amount of change. DPT: 9.001 (temperature (°C))				

381	**Measurement Temperature Internal**	**Temperature Calibration**	**2 bytes**	**CW**

This object is used to calibrate the measurement output by measuring the actual measurement value via an external device and then writing this value to the object. When iX3 received the value, calibrate its measurement output automatically.
*This object is used to calculate the adjustment factor. The adjustment factor can't be zero or negative. Thus, you mustn't enter input value with opposite sign than the current measurement to calculate adjustment factor properly.
DPT: 9.001 (temperature (°C))

382	**Measurement Temperature Internal**	**Alarm - Fault**	**1 bit**	**CT**

This object is used to send an alarm when the sensor is at fault that causes any reason.
DPT: 1.005 (alarm)

383	**Measurement Temperature Internal**	**Alarm - Low**	**1 bit**	**CT**

"Low Level Alarm" object sends "Alarm" telegram when the measurement value goes below the low-level value and "No Alarm" telegram when the measurement value returns above it.
DPT: 1.005 (alarm)

384	**Measurement Temperature Internal**	**Alarm - High**	**1 bit**	**CT**

"Alarm - High" object sends "Alarm" telegram when the measurement value exceeds the high-level value and "No Alarm" telegram when the measurement value returns below it.
DPT: 1.005 (alarm)

385	**Measurement Temperature Internal**	**Additional Value**	**1 bit / 1 bytes**	**CT**

When the measurement value changed, this object sends telegrams with specific type and values according to the related parameters.
DPT: According to parameter selection

5.4.2. Humidity Measurement Objects

Object Number	Object Name	Function	Type	Flags
386	Measurement Humidity Internal	Disable	1 bit	CW

This object is used to set the iX3 internal humidity sensor status. "Enabled" or "Disabled" telegram is received via this object.
For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the internal humidity sensor will continue the measurement. On disabled, any telegram isn't transmitted to the KNX bus.
DPT: 1.003 (enable)

387	Measurement Humidity Internal	Status	1 bit	CRT

This object is used to watch internal humidity sensor status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when internal humidity sensor status is changed over device.
DPT: 1.003 (enable)

388	Measurement Humidity Internal	Humidity Value	2 bytes	CRT

This object is used to send the measurement value that is measured by the sensor after calibrating it. Each measurement value can be calibrated via "Adjustment factor" parameter or "Calibration" object. Depending on the parameter configuration, the calculated data can be sent to the bus line periodically or according to the amount of change.
DPT: 9.007 (humidity (%))

389	Measurement Humidity Internal	Humidity Calibration	2 bytes	CW

This object is used to calibrate the measurement output by measuring the actual measurement value via an external device and then writing this value to the object. When iX3 received the value, calibrate its measurement output automatically.
*This object is used to calculate the adjustment factor. The adjustment factor can't be zero or negative. Thus, you mustn't enter input value with opposite sign than the current measurement to calculate adjustment factor properly.
DPT: 9.007 (humidity (%))

390	**Measurement Humidity Internal**	**Alarm - Fault**	**1 bit**	**CT**

This object is used to send an alarm when the sensor is at fault that causes any reason.
DPT: 1.005 (alarm)

391	**Measurement Humidity Internal**	**Alarm - Low**	**1 bit**	**CT**

"Low Level Alarm" object sends "Alarm" telegram when the measurement value goes below the low-level value and "No Alarm" telegram when the measurement value returns above it.
DPT: 1.005 (alarm)

392	**Measurement Humidity Internal**	**Alarm - High**	**1 bit**	**CT**

"Alarm - High" object sends "Alarm" telegram when the measurement value exceeds the high-level value and "No Alarm" telegram when the measurement value returns below it.
DPT: 1.005 (alarm)

393	**Measurement Humidity Internal**	**Additional Value**	**1 bit / 1 bytes**	**CT**

When the measurement value changed, this object sends telegrams with specific type and values according to the related parameters.
DPT: According to parameter selection

5.4.3. Air Quality Measurement Objects

Object Number	Object Name	Function	Type	Flags
394	Measurement Air Quality Internal	Disable	1 bit	CW

This object is used to set the iX3 internal air quality sensor status. "Enabled" or "Disabled" telegram is received via this object.
For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the internal air quality sensor will continue the measurement. On disabled, any telegram isn't transmitted to the KNX bus.
DPT: 1.003 (enable)

Object Number	Object Name	Function	Type	Flags
395	Measurement Air Quality Internal	Status	1 bit	CRT

This object is used to watch internal air quality sensor status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when internal air quality sensor status is changed over device.
DPT: 1.003 (enable)

Object Number	Object Name	Function	Type	Flags
396	Measurement Air Quality Internal	Air Quality Value	2 bytes	CRT

This object is used to send the measurement value that is measured by the sensor after calibrating it. Each measurement value can be calibrated via "Adjustment factor" parameter or "Calibration" object. Depending on the parameter configuration, the calculated data can be sent to the bus line periodically or according to the amount of change.
DPT: 9.008 (parts/million (ppm))

Object Number	Object Name	Function	Type	Flags
397	Measurement Air Quality Internal	Air Quality Calibration	2 bytes	CW

This object is used to calibrate the measurement output by measuring the actual measurement value via an external device and then writing this value to the object. When iX3 received the value, calibrate its measurement output automatically.
*This object is used to calculate the adjustment factor. The adjustment factor can't be zero or negative. Thus, you mustn't enter input value with opposite sign than the current measurement to calculate adjustment factor properly.
DPT: 9.008 (parts/million (ppm))

398	**Measurement Air Quality Internal**	**Alarm - Fault**	**1 bit**	**CT**

This object is used to send an alarm when the sensor is at fault that causes any reason.
DPT: 1.005 (alarm)

399	**Measurement Air Quality Internal**	**Alarm - Low**	**1 bit**	**CT**

"Low Level Alarm" object sends "Alarm" telegram when the measurement value goes below the low-level value and "No Alarm" telegram when the measurement value returns above it.
DPT: 1.005 (alarm)

400	**Measurement Air Quality Internal**	**Alarm - High**	**1 bit**	**CT**

"Alarm - High" object sends "Alarm" telegram when the measurement value exceeds the high-level value and "No Alarm" telegram when the measurement value returns below it.
DPT: 1.005 (alarm)

401	**Measurement Air Quality Internal**	**Additional Value**	**1 bit / 1 bytes**	**CT**

When the measurement value changed, this object sends telegrams with specific type and values according to the related parameters.
DPT: According to parameter selection

5.4.4. Brightness Measurement Objects

Object Number	Object Name	Function	Type	Flags
402	Measurement Brightness Internal	Disable	1 bit	CW

This object is used to set the iX3 internal brightness sensor status. "Enabled" or "Disabled" telegram is received via this object.
For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the internal brightness sensor will continue the measurement. On disabled, any telegram isn't transmitted to the KNX bus.
DPT: 1.003 (enable)

403	Measurement Brightness Internal	Status	1 bit	CRT

This object is used to watch internal brightness sensor status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when internal brightness sensor status is changed over device.
DPT: 1.003 (enable)

404	Measurement Brightness Internal	Brightness Value	2 bytes	CRT

This object is used to send the measurement value that is measured by the sensor after calibrating it. Each measurement value can be calibrated via "Adjustment factor" parameter or "Calibration" object. Depending on the parameter configuration, the calculated data can be sent to the bus line periodically or according to the amount of change.
DPT: 9.004 (lux)

405	Measurement Brightness Internal	Brightness Calibration	2 bytes	CW

This object is used to calibrate the measurement output by measuring the actual measurement value via an external device and then writing this value to the object. When iX3 received the value, calibrate its measurement output automatically.
*This object is used to calculate the adjustment factor. The adjustment factor can't be zero or negative. Thus, you mustn't enter input value with opposite sign than the current measurement to calculate adjustment factor properly.
DPT: 9.004 (lux)

406	**Measurement Brightness Internal**	**Alarm - Fault**	**1 bit**	**CT**

This object is used to send an alarm when the sensor is at fault that causes any reason.
DPT: 1.005 (alarm)

407	**Measurement Brightness Internal**	**Alarm - Low**	**1 bit**	**CT**

"Low Level Alarm" object sends "Alarm" telegram when the measurement value goes below the low-level value and "No Alarm" telegram when the measurement value returns above it.
DPT: 1.005 (alarm)

408	**Measurement Brightness Internal**	**Alarm - High**	**1 bit**	**CT**

"Alarm - High" object sends "Alarm" telegram when the measurement value exceeds the high-level value and "No Alarm" telegram when the measurement value returns below it.
DPT: 1.005 (alarm)

409	**Measurement Brightness Internal**	**Additional Value**	**1 bit / 1 bytes**	**CT**

When the measurement value changed, this object sends telegrams with specific type and values according to the related parameters.
DPT: According to parameter selection

5.4.6. External Measurement Objects

X: 1 / 2

Object Number	Object Name	Function	Type	Flags
410, 418	**Measurement External X**	**Disable**	**1 bit**	**CW**

This object is used to set the iX3 external X sensor status. "Enabled" or "Disabled" telegram is received via this object.
For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the external X sensor will continue the measurement. On disabled, any telegram isn't transmitted to the KNX bus.
DPT: 1.003 (enable)

411, 419	**Measurement External X**	**Status**	**1 bit**	**CRT**

This object is used to watch external X sensor status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when external X sensor status is changed over device.
DPT: 1.003 (enable)

412, 420	**Measurement External X**	**Temperature Value / Brightness Value**	**2 bytes**	**CRT**

This object is used to send the measurement value that is measured by the sensor after calibrating it. Each measurement value can be calibrated via "Adjustment factor" parameter or "Calibration" object.
DPT: 9.001 (temperature (°C)) / DPT: 9.004 (lux)

413, 421	**Measurement External X**	**Temperature Calibration / Brightness Calibration**	**2 bytes**	**CW**

This object is used to calibrate the measurement output by measuring the actual measurement value via an external device and then writing this value to the object. When iX3 received the value, calibrate its measurement output automatically.
*This object is used to calculate the adjustment factor. The adjustment factor can't be zero or negative. Thus, you mustn't enter input value with opposite sign than the current measurement to calculate adjustment factor properly.
DPT: 9.001 (temperature (°C)) / DPT: 9.004 (lux)

414, 422	**Measurement External X**	**Alarm - Fault**	**1 bit**	**CT**

This object is used to send an alarm when the sensor is at fault that causes any reason.
DPT: 1.005 (alarm)

415, 423	**Measurement External X**	**Alarm - Low**	**1 bit**	**CT**

"Low Level Alarm" object sends "Alarm" telegram when the measurement value goes below the low-level value and "No Alarm" telegram when the measurement value returns above it.
DPT: 1.005 (alarm)

416, 424	**Measurement External X**	**Alarm - High**	**1 bit**	**CT**

"Alarm - High" object sends "Alarm" telegram when the measurement value exceeds the high-level value and "No Alarm" telegram when the measurement value returns below it.
DPT: 1.005 (alarm)

417, 425	**Measurement External X**	**Additional Value**	**1 bit / 1 bytes**	**CT**

When the measurement value changed, this object sends telegrams with specific type and values according to the related parameters.
DPT: According to parameter selection

5.5. Calculation Objects

In this section, Calculation X objects are described in the table below. In the first column name of the object, in the second column function name, the third column data type and fourth column the objects flags, information is given.

X: 1 … 6

Object Number	Object Name	Function	Type	Flags
426, 434 442, 450 458, 466	**Calculation X**	**Disable**	**1 bit**	**CW**
This object is used to set the iX3 calculation X status. "Enabled" or "Disabled" telegram is received via this object. For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the + calculation X will continue the calculation. On disabled, any telegram isn't transmitted to the KNX bus. DPT: 1.003 (enable)				
427, 435 443, 451 459, 467	**Calculation X**	**Status**	**1 bit**	**CRT**
This object is used to watch calculation X status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when calculation X status is changed over device. DPT: 1.003 (enable)				
428, 436 444, 452 460, 468	**Calculation X**	**Probe Input Temperature / Probe Input Humidity / Probe Input Brightness / Probe Input Proximity / Probe Input Air Quality / Probe Input Pressure / Probe Input Wind Speed**	**1 bit / 2 bytes**	**CW**
This object is used to receive the sensor value from the KNX bus line. This value can be used as a single sensor source or mixing part for the value calculation. DPT: According to parameter selection				
429, 437 445, 453 461, 469	**Calculation X**	**Probe Surveillance**	**1 bit**	**CRT**
This object is used to send alarm if new value is not received a long time set "KNX probe surveillance time" parameter. DPT: 1.018 (occupancy)				

430, 438 446, 454 462, 470	**Calculation X**	**Output Temperature / Output Humidity / Output Brightness / Output Proximity / Output Air Quality / Output Pressure / Output Wind Speed**	**2 bytes**	**CRT**

This object is used to send the calculation output value that is calculated by the calculation X channel. Depending on the parameter configuration, the calculated data can be sent to the bus line periodically or according to the amount of change.
DPT: According to parameter selection

431, 439 447, 455 463, 471	**Calculation X**	**Alarm - Low**	**1 bit**	**CRT**

"Low Level Alarm" object sends "Alarm" telegram when the calculation output value goes below the low-level value and "No Alarm" telegram when the calculation output value returns above it.
DPT: 1.005 (alarm)

432, 440 448, 456 464, 472	**Calculation X**	**Alarm - High**	**1 bit**	**CRT**

"Alarm - High" object sends "Alarm" telegram when the calculation output value exceeds the high-level value and "No Alarm" telegram when the calculation output value returns below it.
DPT: 1.005 (alarm)

5.6. Thermostat Objects

In this section, Thermostat objects are described in the table below. In the first column name of the object, in the second column function name, the third column data type and fourth column the objects flags, information is given.

X: 1 … 2, **Y:** 1 … 5, **Z:** 1 / 2

Object Number	Object Name	Function	Type	Flags
474, 545	**Thermostat X**	**Thermostat Disabling**	**1 bit**	**CW / CRT***
This object is used to set the iX3 thermostat status. "Enabled" or "Disabled" telegram is received via this object. For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the iX3 thermostat will continue working. *This object is used as feedback object in thermostat slave mode. DPT: 1.003 (enable)				
475, 546	**Thermostat X**	**Thermostat Status**	**1 bit**	**CRT / CW***
This object is used to watch thermostat status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when thermostat status is changed over device. *This object is used as input object in thermostat slave mode. DPT: 1.003 (enable)				
476, 547	**Thermostat X**	**Thermostat Switch**	**1 bit**	**CRWTU**
This object is used to send on/off value from thermostat control page by pressing ON/OFF icon. It is used for on/off the room controller actuators etc. If this object is OFF, "Thermostat Status" can be "Enabled" but thermostat controller output is OFF. DPT: 1.001 (switch)				
478, 549	**Thermostat X**	**Thermostat Operation Mode**	**1 byte**	**CW / CRT***
This object switches over the operating modes with a 1-byte value. *This object is used as feedback object in thermostat slave mode. DPT: 20.102 (HVAC mode)				

479, 550	Thermostat X	Thermostat Operation Mode Forced	1 byte	CW

This object is used to set operation mode of thermostat. Its priority is highest including thermostat energy saving functions except window contact and the mode cannot be changed until "Auto" is received via this object. If "Auto" is received, the operation mode is back the HVAC mode that before enter the forced operation mode.
DPT: 20.102 (HVAC mode)

480, 551	Thermostat X	Thermostat Operation Mode Status / Operation Mode Feedback	1 byte	CRT / CW*

This object indicates the status of the operating mode with a 1-byte value.
*This object is used as input object in thermostat slave mode.
DPT: 20.102 (HVAC mode)

481, 552	Thermostat X	Operation Mode [Comfort]	1 bit	CW

The Comfort mode activation command is sent via this object. If "On" telegram is received via this object, operation mode is changed as Comfort. If active operation mode is Comfort and "Off" telegram is received via this object, the operating mode is changed as Auto. If weekly program isn't active, the operating mode isn't changed and keep current state.
DPT: 1.001 (switch)

482, 553	Thermostat X	Operation Mode [Standby]	1 bit	CW

The Standby mode activation command is sent via this object. If "On" telegram is received via this object, operation mode is changed as Standby. If active operation mode is Standby and "Off" telegram is received via this object, the operating mode is changed as Auto. If weekly program isn't active, the operating mode isn't changed and keep current state.
DPT: 1.001 (switch)

483, 554	Thermostat X	Operation Mode [Economy]	1 bit	CW

The Economy mode activation command is sent via this object. If "On" telegram is received via this object, operation mode is changed as Economy. If active operation mode is Economy and "Off" telegram is received via this object, the operating mode is changed as Auto. If weekly program isn't active, the operating mode isn't changed and keep current state.
DPT: 1.001 (switch)

484, 555	Thermostat X	Operation Mode [Protection]	1 bit	CW

The Protection mode activation command is sent via this object. If "On" telegram is received via this object, operation mode is changed as Protection. If active operation mode is Protection and "Off" telegram is received via this object, the operating mode is changed as Auto. If weekly program isn't active, the operating mode isn't changed and keep current state.
DPT: 1.001 (switch)

485, 556	**Thermostat X**	**Thermostat Heating/Cooling Switchover**	**1 bit**	**CW / CRT***

This object is used to change over the heating/cooling modes.
*This object is used as feedback object in thermostat slave mode.
DPT: 1.100 (cooling/heating)

486, 557	**Thermostat X**	**Thermostat Heating/Cooling Status**	**1 bit**	**CRT / CW***

Heating/cooling status information is indicated via this object.
*This object is used as input object in thermostat slave mode.
DPT: 1.100 (cooling/heating)

487, 558	**Thermostat X**	**Thermostat Heating Control Disabling**	**1 bit**	**CW**

This object activates or deactivates the heating system.
DPT: 1.003 (enable)

488, 559	**Thermostat X**	**Thermostat Heating Control Running**	**1 bit**	**CRT / CW***

This object is used to inform about the heating control. If the heating control is active and the control value is higher than zero, ON telegram is transmitted to KNX bus. If the heating control is not active and the control value is zero, OFF telegram is transmitted to KNX bus.
*This object is used as input object in thermostat slave mode.
DPT: 1.002 (boolean)

489, 560	**Thermostat X**	**Thermostat Heating Value - Thermostat Heating/Cooling Value**	**1 bit / 1 byte**	**CRT**

The output value of thermostat control is transmitted via the object.
DPT: 1.001 (switch) / 5.004 (percentage (0...255%))

490, 561	**Thermostat X**	**Thermostat Heating Value Request / Heating/Cooling Value Request**	**1 bit**	**CW**

This object is used to get the output value of heating controller. If "Acknowledge command" telegram is received via this object, current value of the heating controller is transmitted to KNX bus.
DPT: 1.016 (acknowledge)

491, 562	**Thermostat X**	**Thermostat Cooling Control Disabling**	**1 bit**	**CW**

This object activates or deactivates the cooling system.
DPT: 1.003 (enable)

492, 563	**Thermostat X**	**Thermostat Cooling Control Running**	**1 bit**	**CRT / CW***

This object is used to inform about the cooling control. If the cooling control is active and the control value is higher than zero, ON telegram is transmitted to KNX bus. If the cooling control is not active and the control value is zero, OFF telegram is transmitted to KNX bus.
*This object is used as input object in thermostat slave mode.
DPT: 1.002 (boolean)

493, 564	**Thermostat X**	**Thermostat Cooling Value**	**1 bit / 1 byte**	**CRT**

The output value of thermostat cooling control is transmitted via the object.
DPT: 1.001 (switch) / 5.004 (percentage (0...255%))

494, 565	**Thermostat X**	**Thermostat Cooling Value Request**	**1 bit**	**CW**

This object is used to get the output value of cooling controller. If "Trigger" telegram is received via this object, current value of the heating controller is transmitted to KNX bus.
DPT: 1.016 (acknowledge)

495, 566	**Thermostat X**	**Thermostat Additional Heating Control Disabling**	**1 bit**	**CW**

This object activates or deactivates the additional heating system.
DPT: 1.003 (enable)

496, 567	**Thermostat X**	**Thermostat Additional Heating Control Running**	**1 bit**	**CRT**

This object is used to inform about the additional heating control. If the additional heating control is active and the control value is higher than zero, ON telegram is transmitted to KNX bus. If the additional heating control is not active and the control value is zero, OFF telegram is transmitted to KNX bus.
DPT: 1.002 (boolean)

497, 568	**Thermostat X**	**Thermostat Additional Heating Value**	**1 bit / 1 byte**	**CRT**

The output value of thermostat additional heating control is transmitted via the object.
DPT: 1.001 (switch) / 5.004 (percentage (0...255%))

498, 569	**Thermostat X**	**Thermostat Additional Heating Value Request**	**1 bit**	**CW**

This object is used to get the output value of additional heating controller. If "Trigger" telegram is received via this object, current value of the heating controller is transmitted to KNX bus.
DPT: 1.016 (acknowledge)

499, 570	**Thermostat X**	**Thermostat Additional Cooling Control Disabling**	**1 bit**	**CW**

This object activates or deactivates the additional cooling system.
DPT: 1.003 (enable)

500, 571	**Thermostat X**	**Thermostat Additional Cooling Control Running**	**1 bit**	**CRT**

This object is used to inform about the additional cooling control. If the additional cooling control is active and the control value is higher than zero, ON telegram is transmitted to KNX bus. If the additional cooling control is not active and the control value is zero, OFF telegram is transmitted to KNX bus.
DPT: 1.002 (boolean)

501, 572	**Thermostat X**	**Thermostat Additional Cooling Value**	**1 bit / 1 byte**	**CRT**

The output value of thermostat additional cooling control is transmitted via the object.
DPT: 1.001 (switch) / 5.004 (percentage (0...255%))

502, 573	**Thermostat X**	**Thermostat Additional Cooling Value Request**	**1 bit**	**CW**

This object is used to get the output value of additional cooling controller. If "Trigger" telegram is received via this object, current value of the heating controller is transmitted to KNX bus.
DPT: 1.016 (acknowledge)

503, 574	**Thermostat X**	**Room Temperature Output (C) - Room Temperature Output (F)**	**1 bit**	**CRT / CW***

This object is used to inform about the temperature value that room controller uses.
*This object is used as input object if thermostat temperature source is selected as "Temperature object".
DPT: 9.001 (temperature (°C)) / 9.027 (temperature difference (K))

504, 575	**Thermostat X**	**Actual Setpoint Output**	**2 bytes**	**CRT / CW***

The pre-configured setpoint temperature is obtained with this object.
*This object is used as input object in thermostat slave mode.
DPT: According to parameter selection

505, 576	Thermostat X	Manual Setpoint Input	2 bytes	CW / CRT*

The setpoint temperature is configured manually with this object. If HVAC mode is Build Protection, the setpoint can't be changed via this object.
If the difference between the active setpoint and received value is higher than the "Manual setpoint range" parameter, Manual Setpoint value is set maximum or minimum limit value according to "Manual setpoint range" parameter.
*This object is used as feedback object in thermostat slave mode.
DPT: According to parameter selection

506, 577	Thermostat X	Manual Setpoint Reset	1 bit	CW

The setpoint temperature that is desired to configure manually can be reset with this object.
DPT: 1.015 (reset)

507, 578	Thermostat X	Heating Comfort Setpoint Temperature	2 bytes	CW

The setpoint temperature value for heating comfort mode is configured with this object.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

508, 579	Thermostat X	Heating Standby Setpoint Temperature	2 bytes	CW

The setpoint temperature value for heating standby mode is configured with this object.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

509, 580	Thermostat X	Heating Economy Setpoint Temperature	2 bytes	CW

The setpoint temperature value for heating economy mode is configured with this object.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

510, 581	Thermostat X	Heating Protection Setpoint Temperature	2 bytes	CW

The setpoint temperature value for heating protection mode is configured with this object.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

511, 582	Thermostat X	Cooling Comfort Setpoint Temperature	2 bytes	CW

The setpoint temperature value for cooling comfort mode is configured with this object.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

512, 583	Thermostat X	Cooling Standby Setpoint Temperature	2 bytes	CW

The setpoint temperature value for cooling standby mode is configured with this object.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

513, 584	Thermostat X	Cooling Economy Setpoint Temperature	2 bytes	CW

The setpoint temperature value for cooling economy mode is configured with this object.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

514, 585	Thermostat X	Cooling Protection Setpoint Temperature	2 bytes	CW

The setpoint temperature value for cooling protection mode is configured with this object.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

515, 586	Thermostat X	Fan Controller Disable	1 bit	CW / CRT*

This object is used to set the iX3 fan controller status. "Enabled" or "Disabled" telegram is received via this object.
For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the iX3 fan controller will continue working.
*This object is used as feedback object in thermostat slave mode.
DPT: 1.003 (enable)

516, 587	Thermostat X	Fan Controller Status	1 bit	CWT / CW*

This object is used to watch fan controller status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when fan controller status is changed over device.
*This object is used as input object in thermostat slave mode.
DPT: 1.003 (enable)

517, 588	Thermostat X	Fan Controller Working Mode	1 bit	CW / CRT*

This object is used to switch over to automatic or manual fan speed control mode.
*This object is used as feedback object in thermostat slave mode.
DPT: 1.001 (switch)

518, 589	Thermostat X	Fan Controller Working Mode Status	1 bit	CRT / CW*

This object indicates the manual / automatic fan operating mode with 1 bit value.
*This object is used as input object in thermostat slave mode.
DPT: 1.001 (switch)

519, 590	Thermostat	Fan Controller Proportional Output	1 byte	CRT

This object is used to send the output value of the fan proportional controller.
DPT: 5.001(percentage (0...100%))

520, 591	**Thermostat**	**Fan Controller Manual Step / Fan Controller Manual Up/Down**	**1 bit**	**CW**

This object is used to increase or decrease the fan speed
DPT: 1.007 (step) / 1.008 (up/down)

521, 592	**Thermostat**	**Fan Controller Manual Stage**	**1 byte**	**CW / CRT***

This object allows the manual fan speed to be controlled with 1-byte value.
*This object is used as feedback object in thermostat slave mode.
DPT: 5.100(fan stage (0...255))

522, 593	**Thermostat X**	**Fan Controller Speed (1 Byte)**	**1 byte**	**CRT**

This object allows the fan speed to be controlled with 1-byte value.
DPT: 5.010 (counter pulses (0...255))

523, 594	**Thermostat X**	**Fan Controller Speed Feedback Input (1 Byte)**	**1 byte**	**CWU**

This object waits the fan speed feedback with a 1-byte value.
DPT: 5.010 (counter pulses (0...255))

524, 595 525, 596 526, 597 527, 598 528, 599	**Thermostat X**	**Fan Level Y**	**1 bit**	**CRT**

This object indicates the Fan Level Y value with a 1-bit value.
DPT: 1.001 (switch)

529, 600 530, 601 531, 602 532, 603 533, 604	**Thermostat X**	**Fan Level Y Feedback Input**	**1 bit**	**CWU**

This object indicates the Fan Level X status with a 1-bit value.
DPT: 1.001 (switch)

534, 605 535, 606	**Thermostat X**	**Energy Saving – Window Contact Z**	**1 bit**	**CW**

This object is used to activate window contact function.
DPT: 1.001 (switch)

536, 607 **537, 608**	**Thermostat X**	**Energy Saving – Presence Input Z**	**1 bit**	**CW**

This object is used to activate presence input function.
DPT: 1.001 (switch)

538, 609 **539, 610**	**Thermostat X**	**Energy Saving – Card Holder Z**	**1 bit**	**CW**

This object is used to activate card holder function.
DPT: 1.001 (switch)

540, 611	**Thermostat X**	**Temperature Limit Heating Source**	**2 bytes**	**CW**

This group object receives the limit temperature for heating stage. The temperature value received here is used to evaluate the limit temperature. The limit becomes active when the temperature set in the parameter is exceeded.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

541, 612	**Thermostat X**	**Temperature Limit Cooling Source**	**2 bytes**	**CW**

This group object receives the limit temperature for cooling stage. The temperature value received here is used to evaluate the limit temperature. The limit becomes active when the temperature set in the parameter is fallen below.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

542, 613	**Thermostat X**	**Temperature Limit Additional Heating Source**	**2 bytes**	**CW**

This group object receives the limit temperature for additional heating stage. The temperature value received here is used to evaluate the limit temperature. The limit becomes active when the temperature set in the parameter is exceeded.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

543, 614	**Thermostat X**	**Temperature Limit Additional Cooling Source**	**2 bytes**	**CW**

This group object receives the limit temperature for additional cooling stage. The temperature value received here is used to evaluate the limit temperature. The limit becomes active when the temperature set in the parameter is fallen below.
DPT: 9.001 (temperature (°C)) / DPT: 9.027 (temperature (°F))

5.7. Air Conditioner Objects

In this section, Air Conditioner objects are described in the table below. In the first column name of the object, in the second column function name, the third column data type and fourth column the objects flags, information is given.

X: 1 … 2, **Y:** 1 … 7, **Z:** Auto, 0 … 5

Object Number	Object Name	Function	Type	Flags
616, 657	**Air Conditioner X**	**Disabling**	**1 bit**	**CW**
This object is used to set the iX3 air conditioner status. "Enabled" or "Disabled" telegram is received via this object. For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the iX3 air conditioner will continue working. DPT: 1.003 (enable)				
617, 658	**Air Conditioner X**	**Status**	**1 bit**	**CRT**
This object is used to watch air conditioner status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when air conditioner status is changed over device. DPT: 1.003 (enable)				
618, 659	**Air Conditioner X**	**Switch**	**1 bit**	**CRT**
This object is used to send on/off value from air conditioner control page by pressing ON/OFF icon. It is used for on/off the AC device etc. If this object is OFF, AC channel can be sent the object to KNX bus. DPT: 1.001 (switch)				
619, 660	**Air Conditioner X**	**Switch Feedback**	**1 bit**	**CW**
This object is used to watch status of "AC Switch" object. AC channel makes correction via this object. DPT: 1.001 (switch)				
620, 661	**Air Conditioner X**	**Room Temperature Input**	**2 bytes**	**CW**
This object is enabled if AC temperature source is selected as "temperature object". AC channel read room temperature via this object. DPT: 9.001 (temperature (°C))				

621, 662	**Air Conditioner X**	**Room Temperature Output**	**2 bytes**	**CRT**

This object is used to inform about the temperature value that air conditioner uses.
DPT: 9.001 (temperature (°C))

622, 663	**Air Conditioner X**	**Setpoint Temperature**	**2 bytes**	**CRT**

This object is used to send the setpoint temperature to AC device.
DPT: 9.001 (temperature (°C))

623, 664	**Air Conditioner X**	**Mode**	**1 byte**	**CRT**

This object is used to send telegram of each mode of AC device.
DPT: 20.105 (HVAC control mode)

624, 665	**Air Conditioner X**	**Mode Feedback**	**1 byte**	**CW**

This object is used to receive the status telegram of each mode of AC device.
DPT: 20.105 (HVAC control mode)

625, 666	**Air Conditioner X**	**Mode Auto**	**1 bit**	**CRWTU**

This object is used to send the mode auto control telegram and also receive status feedback.
DPT: 1.003 (enable)

626, 667	**Air Conditioner X**	**Mode Heat**	**1 bit**	**CRWTU**

This object is used to send the mode heating control telegram and also receive status feedback.
DPT: 1.003 (enable)

627, 668	**Air Conditioner X**	**Mode Cool**	**1 bit**	**CRWTU**

This object is used to send the mode cool control telegram and also receive status feedback.
DPT: 1.003 (enable)

628, 669	**Air Conditioner X**	**Mode Dry**	**1 bit**	**CRWTU**

This object is used to send the mode dry control telegram and also receive status feedback.
DPT: 1.003 (enable)

629, 670	**Air Conditioner X**	**Mode Fan**	**1 bit**	**CRWTU**

This object is used to send the mode fan control telegram and also receive status feedback.
DPT: 1.003 (enable)

630, 671	**Air Conditioner X**	**Extension Mode**	**1 byte**	**CRT**

This object is used to send telegram of extension modes of AC device. These modes are user defined modes.
DPT: 1 byte

631, 672	**Air Conditioner X**	**Extension Mode Feedback**	**1 byte**	**CW**

This object is used to receive the status telegram of extension modes of AC device.
DPT: 1 byte

632, 633, 634 635, 636, 637 638, 673, 674 678, 679	**Air Conditioner X**	**Extension Mode Y**	**1 bit**	**CRWTU**

This object is used to send the extension mode Y control telegram and also receive status feedback.
DPT: 1.003 (enable)

639, 680	**Air Conditioner X**	**Fan Level**	**1 byte**	**CRT**

This object is used to send telegram of fan speed of AC device.
DPT: 5.100 (fan stage (0-255))

640, 681	**Air Conditioner X**	**Fan Level Feedback**	**1 byte**	**CW**

This object is used to receive the status telegram of fan speed of AC device.
DPT: 5.100 (fan stage (0-255))

641, 642, 643, 644, 645, 646, 682, 683, 684, 685, 686, 687	**Air Conditioner X**	**Fan Level Z**	**1 bit**	**CRWTU**

This object is used to send the fan speed's 1-bit control telegram and also receive status feedback.
DPT: 1.002 (boolean)

647, 688	**Air Conditioner X**	**Fan Swing Vertical Moving**	**1 byte**	**CRWTU**

This object is used to send the movement type of vertical swing such as fixed or moving telegram and also receive status feedback.
DPT: 1.003 (enable)

648, 689	**Air Conditioner X**	**Fan Swing Horizontal Moving**	**1 byte**	**CRWTU**

This object is used to send the movement type of horizontal swing such as fixed or moving telegram and also receive status feedback.
DPT: 1.003 (enable)

649, 690	**Air Conditioner X**	**Fan Level Vertical**	**1 byte**	**CRT**

This object is used to send telegram of vertical swing level of AC device.
DPT: 5.010 (counter pulses (0-255))

650, 691	**Air Conditioner X**	**Fan Level Horizontal**	**1 byte**	**CRT**

This object is used to send telegram of horizontal swing level of AC device.
DPT: 5.010 (counter pulses (0-255))

651, 692	**Air Conditioner X**	**Fan Level Vertical Feedback**	**1 byte**	**CW**

This object is used to receive the status telegram of vertical swing level of AC device.
DPT: 5.010 (counter pulses (0-255))

652, 693	**Air Conditioner X**	**Fan Level Horizontal Feedback**	**1 byte**	**CW**

This object is used to receive the status telegram of horizontal swing level of AC device.
DPT: 5.010 (counter pulses (0-255))

653, 694	**Air Conditioner X**	**Error 2 Byte**	**2 bytes**	**CW**

This object is used to receive the 2-byte error code of AC device.
DPT: 2-byte

654, 695	**Air Conditioner X**	**Error Text**	**14 bytes**	**CW**

This object is used to receive the 14-byte character string error code of AC device.
DPT: 16.000 (Character String (ASCII))

5.10. Additional Functions – Logic Objects

This section contains information about KNX objects and their properties related to the logic function channels. The types, flags and properties of the objects are explained in detail below. There are 8 identical logic channels in the device, so only one logical channel is described here. The X values can be between 1…8 and Y values also can be 1…5. Please do not forget to take this into account.

X: 1 … 8, **Y:** 1 … 5

Object Number	Object Name	Function	Type	Flags
698, 726, 754 782, 810, 838 866, 894	**Logic X**	**Lock**	**1 bit**	**CW**
This object is used to set the logic lock status. "On" or "Off" telegram is received via this object. According to selected parameter in ETS, it will be disabled when an "On" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the logic will continue working or vice versa. DPT: 1.001 (switch)				
699, 727, 755 783, 811, 839 867, 895	**Logic X**	**Status**	**1 bit**	**CRT**
This object is used to watch alarm status. "On" or "Off" telegram is transmitted to KNX bus via this object when alarm status is changed over device. It becomes visible when the "use logic lock" parameter is set to yes. DPT: 1.001 (switch)				
700, 728, 756 784, 812, 840 868, 896	**Logic X**	**External Movement**	**1 bit**	**CWU**
This object is used to receive movement information from the KNX bus line. According to the ETS parameter configuration, the '0' or '1' value is accounted as there is a movement detection occurs. DPT: 1.001 (switch)				
701, 729, 757 785, 813, 841 869, 897	**Logic X**	**External Brightness**	**2 bytes**	**CWU**
This object is used to obtain a brightness value from the KNX bus line. The received brightness value will be used to evaluate the input status according to the brightness thresholds. DPT: 9.004 (lux)				

702, 730, 758 786, 814, 842 870, 898,	**Logic X**	**Brightness Threshold Lower**	**2 bytes**	**CW**

This object is used to receive the brightness threshold lower value from the KNX bus line. The value read on this object is will be used as a new brightness threshold lower value. This object becomes visible when the "Change brightness threshold via bus" parameter is set to yes.
DPT: 9.004 (lux)

703, 731, 759 787, 815, 843 871, 899	**Logic X**	**Brightness Threshold Upper**	**2 bytes**	**CW**

This object is used to receive the brightness threshold upper value from the KNX bus line. The value read on this object is will be used as a new brightness threshold upper value. This object becomes visible when the "Change brightness threshold via bus" parameter is set to yes.
DPT: 9.004 (lux)

704, 732, 760 788, 816, 844 872, 900	**Logic X**	**External Temperature**	**2 bytes**	**CWU**

This object is used to obtain a temperature value from the KNX bus line. The received temperature value will be used to evaluate the input status according to the temperature thresholds.
DPT: 9.001 (temperature)

705, 733, 761 789, 817, 845 873, 901	**Logic X**	**Temperature Threshold Lower**	**2 bytes**	**CW**

This object is used to receive the temperature threshold lower value from the KNX bus line. The value read on this object is will be used as a new temperature threshold lower value. This object becomes visible when the "Change temperature via bus" parameter is set to yes.
DPT: 9.004 (lux)

706, 734, 762 790, 818, 846 874, 902	**Logic X**	**Temperature Threshold Upper**	**2 bytes**	**CW**

This object is used to receive the temperature threshold upper value from the KNX bus line. The value read on this object is will be used as a new temperature threshold upper value. This object becomes visible when the "Change temperature via bus" parameter is set to yes.
DPT: 9.004 (lux)

707, 708, 709 735, 736, 737 763, 764, 765 791, 792, 793 819, 820, 821 847, 848, 849 875, 876, 877 903, 904, 905	**Logic X**	**External Input – 1 / 2 / 3**	**1 bit / 1 byte / 2 byte / 4 byte**	**CWU**

This object is used to obtain external input 1 / 2 / 3 information from the KNX bus line. According to the ETS parameter configuration, the received values are accounted as TRUE or FALSE for this external input. For 1 bit configuration, there is only '1' or '0' values for calculating the input status. But for other input (such as 1 byte, etc.) the received value is compared to the external input value parameter.
DPT: According to parameter selection, DPT changes.

710, 738, 766 794, 822, 850 878, 906	**Logic X**	**Result Status**	**1 bit**	**CRT**

This object is used to send the related logic function block's result status to the KNX bus line. According to the ETS parameter configuration, this value can be sent periodically, on change or only configured value. (TRUE or FALSE).
DPT: 1.002 (boolean)

711, 714, 717 720, 723, 739 742, 745, 748 751, 767, 770 773, 776, 779 795, 798, 801 804, 807, 823 826, 829, 832 835, 851, 854 857, 860, 863 879, 882, 885 888, 891, 907 910, 913, 916 919	**Logic X**	**Output Switch Controller Output Absolute Dimming Controller Output Shutter Controller Output Alarm Controller Output Sequence Controller Output Scene Controller Output String Controller Output Threshold Controller**	**1 bit 1 byte 2 bytes**	**CRT**

This object is used to send the related output object's value to the KNX bus line. When the logic function block's status changes, the sending value also can be configured separately. In addition, according to the output type, the object's value type will be changed.
DPT: According to parameter selection, DPT changes.

712, 715, 718 721, 724, 740 743, 746, 749 752, 768, 771 774, 777, 780 796, 799, 802 805, 808, 824 827, 830, 833 836, 852, 855 858, 861, 864 880, 883, 886 889, 892, 908 911, 914, 917 920	**Logic X**	**Delay Time on True State**	**2 bytes**	**CW**

This object is used to receive the 'delay time on TRUE state' value from the KNX bus line. When a new value is received from this object, the received value is used as the output on delay time for the TRUE state value. The configured parameter value will not be used anymore. This object becomes visible when the "Change on time via bus" parameter is set to Yes.
DPT: 7.005 (time (s))

713, 716, 719 722, 725, 741 744, 747, 750 753, 769, 772 775, 778, 797, 800, 803, 806 809, 825, 828 831, 834, 837 853, 856, 859 862, 865, 881 884, 887, 890, 893, 909, 912 915, 918, 921	**Logic X**	**Delay Time on False State**	**2 bytes**	**CW**

This object is used to receive the 'delay time on FALSE state' value from the KNX bus line. When a new value is received from this object, the received value is used as the output on delay time for the FALSE state value. The configured parameter value will not be used anymore. This object becomes visible when the "Change on time via bus" parameter is set to Yes.
DPT: 7.005 (time (s))

5.11. Additional Functions – Converter Objects

In this section, converter objects are described in the table below. Converter group objects are used to make mathematical operations, data converting from different types. Up to 8 different converters can be configured. In the first column name of the object, in the second column function name, the third column data type and fourth column the objects flags, information is given.

X: 1 … 8

Object Number	Object Name	Function	Type	Flags
922, 933, 944 955, 966, 977 988, 999,	**Converter X**	**Disabling**	**1 bit**	**CW**
This object is used to set the converter status. "Enabled" or "Disabled" telegram is received via this object. For example, it will be disabled when an "Enabled" telegram is received from the KNX bus line, and when a "Disabled" telegram is received, the converter will continue working. DPT: 1.003 (enable)				
923, 934, 945 956, 967, 978 989, 1000	**Converter X**	**Status**	**1 bit**	**CRT**
This object is used to watch converter status. "Enabled" or "Disabled" telegram is transmitted to KNX bus via this object when converter status is changed over device. DPT: 1.003 (enable)				

5.11.1. Converter – Gate Forwarding Objects

X: 1 … 8

Object Number	Object Name	Function	Type	Flags
924, 935, 957 **968, 979, 990** **1001**	**Converter X**	**Input Bit** **Input 2Bit** **Input Byte** **Input 2Bytes**	**1 bit** **2 bits** **1 byte** **2 bytes**	**CW**

This object is used to input a value that needs to be converted.
DPT: According to parameter selection, DPT changes

932, 943, 946 **954, 965, 976** **987, 998, 1009**	**Converter X**	**Output Bit** **Output 2Bit** **Output Byte** **Output 2Bytes**	**1 bit** **2 bits** **1 byte** **2 bytes**	**CRT**

This object is used to output the converted value.
DPT: According to parameter selection, DPT changes

5.11.2. Converter – Format Converter Objects

X: 1 … 8

Object Number	Object Name	Function	Type	Flags
924, 925, 926 927, 928, 929 930, 931, 935 936, 937, 938 939, 940, 941 942, 946, 947 948, 949, 950 951, 952, 953 957, 958, 959 960, 961, 962 963, 964, 968 969, 970, 971 972, 973, 974 975, 979, 980 981, 982, 983 984, 985, 986 990, 991, 992 993, 994, 995 996, 997, 1001 1002, 1003, 1004, 1005, 1006, 1007, 1008	Converter X	Input Bit Input Bit: 0 / 1 / 2 / 3 / 4 / 5 / 6 / 7 Input Byte Input RGB Input RGBW Input Red / Green / Blue / White	1 bit 1 byte 3 bytes 6 bytes	CW

This object is used to input a value that needs to be converted.
DPT: According to parameter selection, DPT changes

925, 926, 927 928, 929, 930 931, 932, 936 937, 938, 939 940, 941, 942 943, 947, 948 949, 950, 951 952, 953, 954 958, 960, 961 962, 963, 964 965, 969, 970 971, 972, 973 974, 975, 976 981, 982, 983 984, 985, 986 987, 991, 992 993, 994, 995 996, 997, 998 1002, 1003, 1004, 1005, 1006, 1007, 1008, 1009	**Converter X**	**Output Bit: 0 / 1 / 2 / 3 / 4 / 5 / 6 / 7 Output Byte Output RGB Output RGBW Output Red / Green / Blue / White**	**1 bit 1 byte 3 bytes 6 bytes**	**CRT**

This object is used to output the converted value.
DPT: According to parameter selection, DPT changes

CONTACT INFORMATION

THE INTERRA WEBSITE

Interra provides documentation support via our website www.interratechnology.com. This website is used as a means to make files and information easily available to customers. Accessible by using your favourite Internet browser, the website contains the following information:

- Information about our products and projects.
- Overview of Interra company and values.
- Product Support: Datasheets, product manuals, application descriptions, latest software releases, ETS databases and archived software.

EUROPE, Turkey

Interra
Cumhuriyet Mah. Kartal Cad. Interra R&D Centre No:95/1 Kartal/İstanbul
Tel: +90 (216) 326 26 40 Fax: +90 (216) 324 25 03
Web address: http://www.interratechnology.com