

# Software Description

Subject to technical alteration

Issue March 2019

## thanos

---

## 1 Overview

The application for room operating panel **thanos** can take over the functions temperature control with PI-controller for heating/cooling, temperature detection, set point adjustment, occupancy key with overtime function and fan stage adjustment. The defaults of the LonMark® function profiles **8500 „Space Comfort Controller“**, **3250 „Scene Panel“**, **3200 „Switch“** und **1 „Open Loop Sensor“** are considered. For extended setting options, user defined configuration properties (UCPT) are used. The UCPTs used are defined in the **Thermokon Device Resource Files** from version 2.5 or higher.

**Temperature measurement:** The measurement is made by an internal sensor or by an external LON sensor via the input variables nviSpaceTemp.

### 1.1 Integration

The device can be commissioned via the Service-Pin. The call-up is made by simultaneous pressing of key 1 and 7 for about 5s. See 2.4.1 Service PIN.

Note: Application thanos\_scc\_01 can only be used using LNS Version 3.20 and higher!

**IMPORTANT:** After the device was parameterized using the Plugin, the network variable nviWriteConfig has to be written once (regardless of which value) to activate the configuration parameters in device!!

## 2 Device description

### 2.1 Operating control surface

For the **thanos** type S / SQ the operating interface is divided into one, and for the L / LQ into two zones:

- Menu area for control and display
- Keypad for control (Version L / LQ only)

On **thanos** S / SQ is also the possibility to configure submenus, over which a similar functionality as the keypad on **thanos** L / LQ is available. For details, see Chapter 3.3

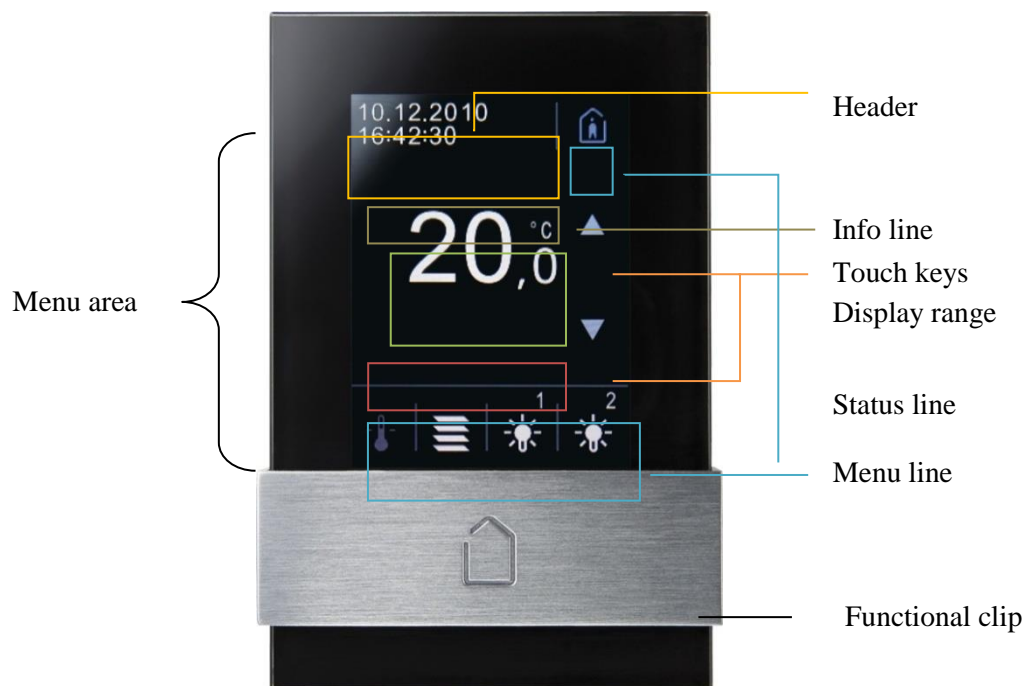


Figure 1 – **thanos** S

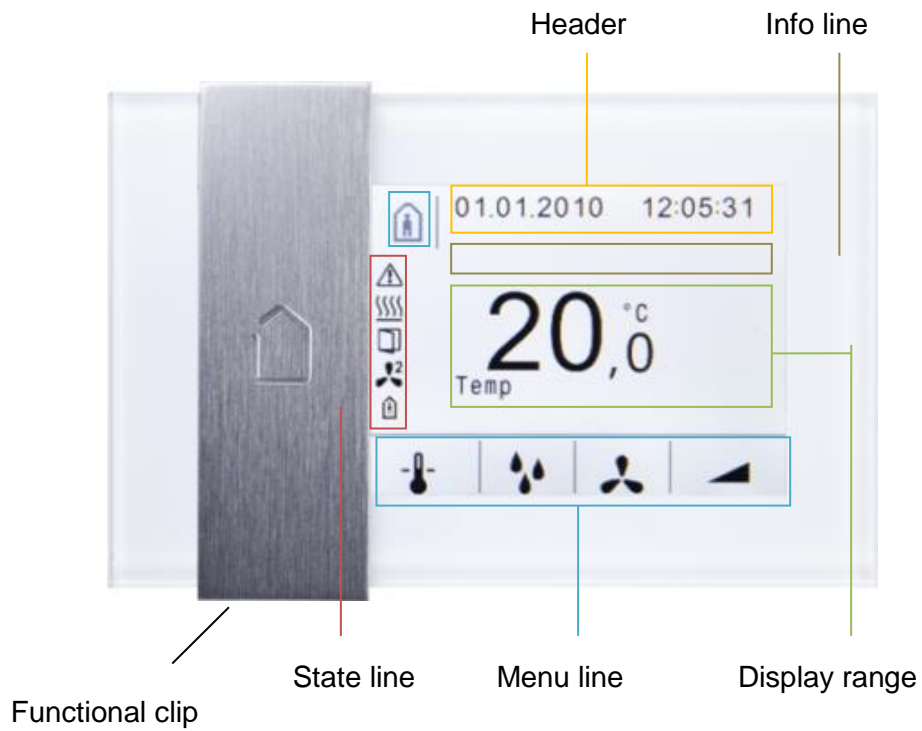


Figure 2– thanos SQ

Device description

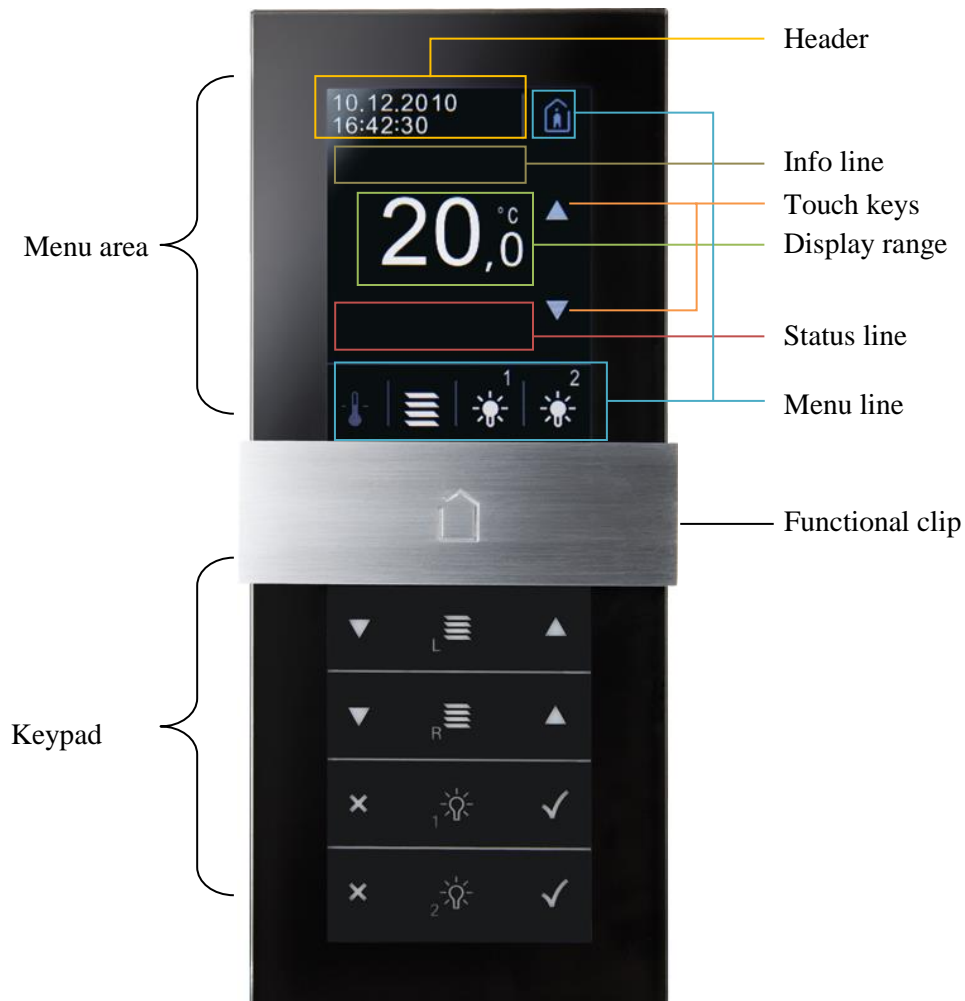


Figure 2-3 thanos L Operating interface

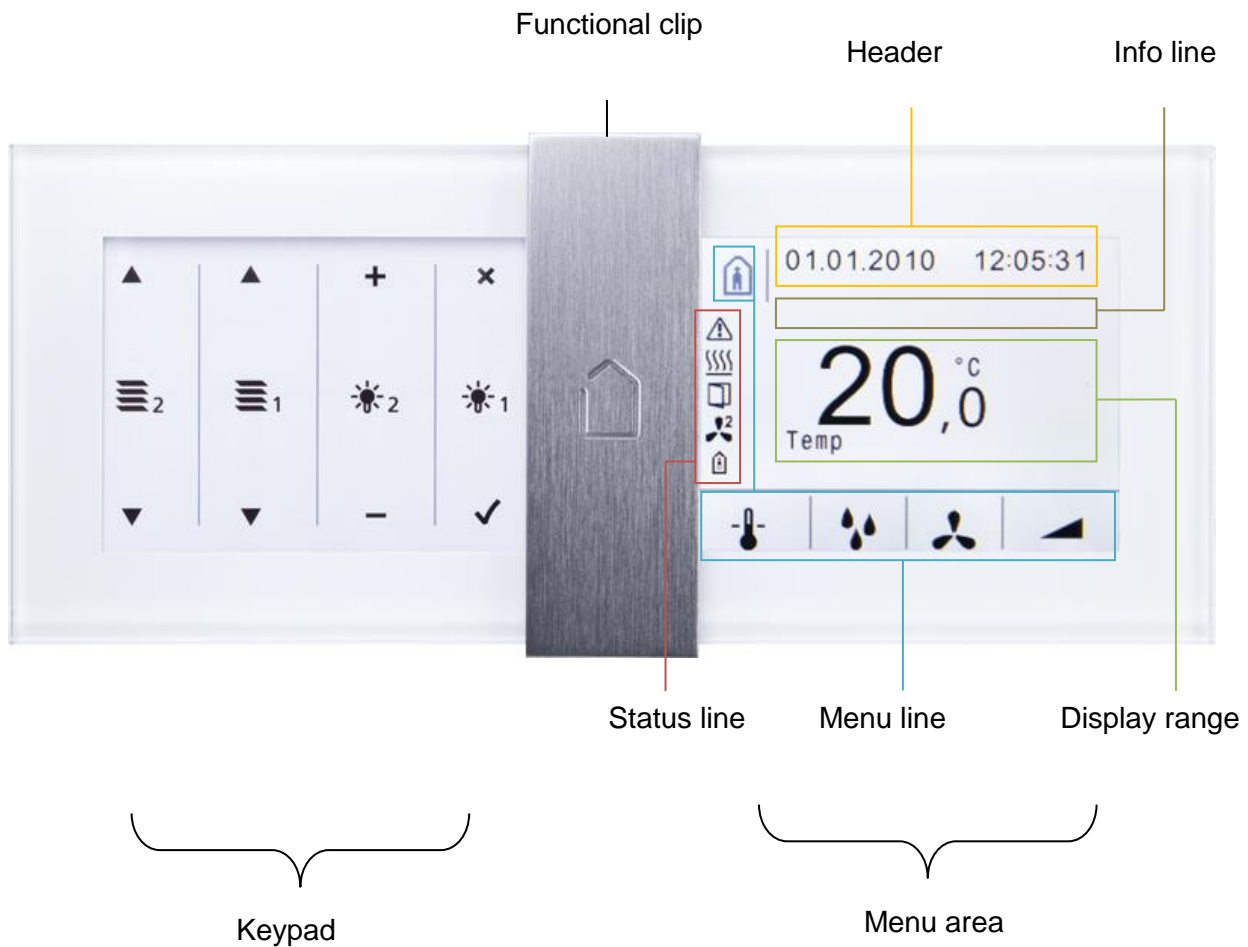


Figure 3-4 - thanos LQ operating interface

The menu area consists of a 3.5" big TFT-Display with 262.144 colors and a touch-display for controlling. Via filed menus in the menu-line different values can be displayed and changed. It can be parameterized which values are displayed and which menu-points are present.

The keypad consists of an individually printable slide-in foil and a touch-surface. The keys of the area can be occupied with several functions.

## Device description

### 2.2 Menu area

#### Headline:

The current date and time can be displayed in different formats and types. The battery-buffered Real Time Clock of the thanos provides for the correct time in case of a voltage reset.

#### Info line:

Arbitrary messages with a length of max. 14 symbols can be displayed in the info line.

#### Display range:

In the display range following functions can be chosen:

- Room temperature, relative humidity (optional)
- 1 Setpoint of integrated controller
- 5 set-points effective and offset with arbitrary unit and description
- 5 external values with arbitrary unit and description

Furthermore values and states of the currently activated menu are displayed.

#### Status line:

In the status line different symbols for fan levels, room occupation, malfunction, heating, cooling, window and dew point can be displayed.



#### Menu line:

In the menu line different menu points can be saved and called by the user through touching the corresponding symbol.

Following menu items can be parameterized:

#### Set point



#### Fan coil



#### Occupancy mode



#### Light, Blind, Universal



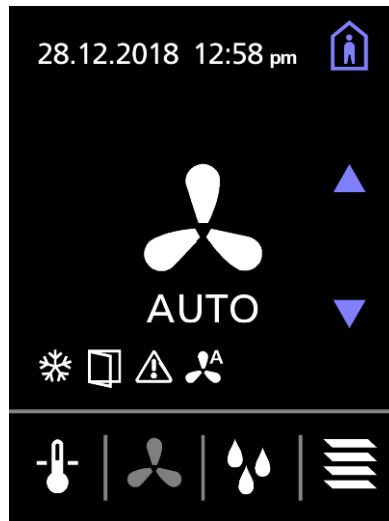
If a menu point is selected, the corresponding symbol is displayed grey-shaded in the menu line and in the display line the value/status is displayed which can be changed in the corresponding menu. By means of the operating key (depending on the function: either ▲ / ▼ or ✓ / ✗) the value/status can be changed afterwards.

Device description

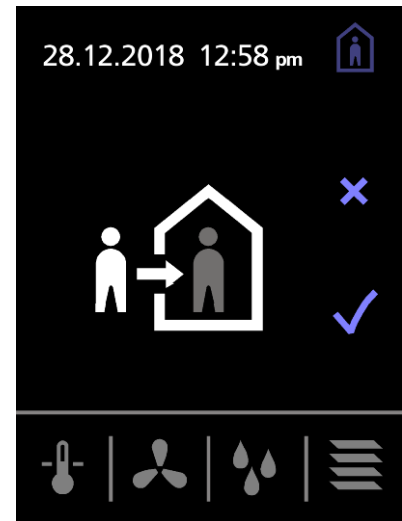
Examples:



Menu „temperature set point“



Menu „fan stages“



Menu „occupancy mode“

Illustration 2-5 Menus

## 2.3 Touch keys

The keypad consists of 8 keys in total. The keys are soft keys, so that the functions of the keys can be freely adjusted via the configuration software. If a key is touched, the corresponding function is visually shown in the display

Example:



Illustration 2-6 Touch keys

In the lower operating interface, the key “blind 2 up” was selected. In the display area the corresponding symbol is displayed in big. Next to it the actuated symbol, e.g. ▲ is displayed. After a freely programmable time, the display indication is reset to the original display indication.

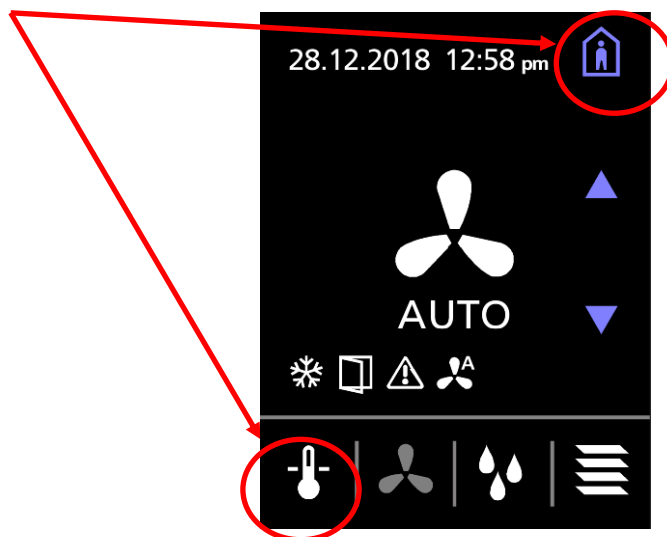
For thanos S/SQ it is possible to parameterize a field programmable sub-menu by which functionality similar to the keypad of thanos L/LQ can be realized.



## 2.4 Startup

For LON-specific properties the thanos has a special menu. The selection is made by simultaneous pressing of keys 1 and 7 for about 5s. The start-up screen at first indication will look like following:

key 1 and 7



### Key functions



scrolling in the menu



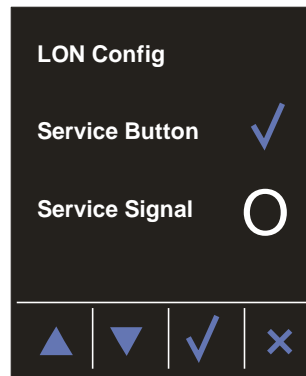
set values



Accept/Quit action. Both actions will exit the configuration menu.

## Device description

Following settings are possible in the modification mode: service PIN



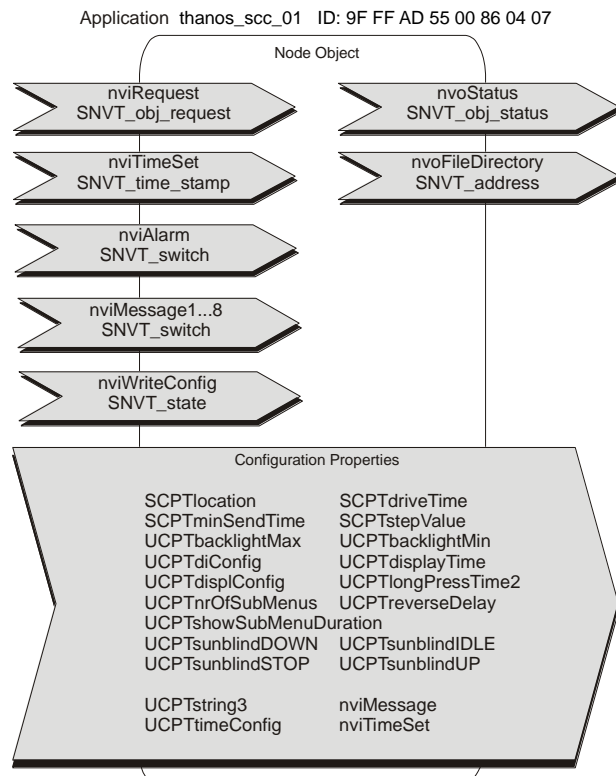
Menu service PIN

### 2.4.1 Service PIN

Service-Pin must be pushed for integration into the network!

### 3 Node Object

The Node Object supervises and controls the functions of the individual objects in the device. The basic functions required by the LonMark® are supported, whereas general network variables and configuration properties for the control and parameterization of the device were added.



#### Time/Date:

Time and date can be shown in the display in field 1 and 2. After a module reset the display for date and time is faded out in the upper field. Only if a message is received via the input variable `nviTimeSet` the time is faded in. The time should be synchronized via a LON system clock once a day. The display of the time can be adapted to individual requests by `UCPTtimeConfig`.

**Display of Measuring Values:** Additionally to the room temperature also other values such as set point, outdoor temperature, CO2 concentration and percentage values can be displayed. Configuration is made via `UCPTdisplConfig` whereas the display is changing between the different readings in the time interval `UCPTdisplayTime`. Measuring values are displayed in field 2.

#### Menu Keys:

By `UCPTdiConfig` the menu keys of the display and the digital inputs including set point adjustment, fan stage adjustment, room occupancy and switch functions can be configured.

#### Background Illumination:

In the stand-by mode, i.e. if no key actuation is made for more than 15 sec., the value `UCPTbacklightMin` determines the brightness of the LCD. Through key actuation the stand-by mode is left and the value `UCPTbacklightMax` determines the brightness of the background illumination.

**IMPORTANT:** After the device was parameterized using the Plugin, the network variable `nviWriteConfig` has to be written once (regardless of which value) to activate the configuration parameters in device!!

#### 3.1 Input Variables Node Object:

##### nviRequest

SNVT Type: SNVT\_obj\_request, Index 92

Function: Input variable including the functions `RQ_NORMAL`, `RQ_UPDATE_STATUS` and `RQ_REPORT_MASK`.

##### nviTimeSet

SNVT Type: SNVT\_time\_stamp, Index 84

Function: Input variable for synchronisation of the time displayed by means of a LON system clock. After a module reset the time is faded out as long as the first valid value is received.

##### nviAlarm

SNVT Type: SNVT\_switch, Index 95

Function: The input variable controls the display symbol „failure“:  
`nviAlarm = 100.0 1 ==> symbol „failure“ on`  
`nviAlarm = 0.0 0 ==> symbol „failure“ off`

## Node Object

### nviMessage1; nviMessage2 ... nviMessage8

SNVT Typ: SNVT\_switch, Index 95

Function: The input variable will be shown in the display depending on the configuration.

Parameter: Using property UCPTString3, text will be indicated.

### nviWriteConfig

SNVT Type: 46, SNVT\_state

Function: After the device was parameterized using the Plugin, the network variable nviWriteConfig has to be written once (regardless of which value) to activate the configuration parameters in device!

## 3.2 Output Variables Node Object:

### nvoStatus

SNVT Type: SNVT\_obj\_status, Index 93

Function: Output variable including the requested status bits „invalid\_id“ and „invalid\_request“.

### nvoFileDirectory

SNVT Type: SNVT\_address, Index 114

Function: The output variable makes the address data of the configuration property in the device available to the LON-integration tool.

## 3.3 Configuration Property Node Object:

### SCPTlocation

SCPT Index: 17, SNVT\_str\_asc

Function: Additional input option to save information on the location in the device.

### UCPTtimeConfig

UCPT Index: 45, typedef struct {unsigned short Byte[4]} UNVT\_str\_hex4

Function: By UCPTtimeConfig the display mode of weekday, date and time can be configured.  
UCPTtimeConfig[0].Byte[0...3] configures time and date in the upper field of the display.  
!! The time should be synchronized by a LON system clock once a day.

UCPTtimeConfig.Byte[0] = 0 ==> <b>Display date</b> faded out	
UCPTtimeConfig.Byte[0] = 1 ==> <b>Display date</b> english	
UCPTtimeConfig.Byte[0] = 2 ==> <b>Display date</b> german	<b>Default</b>
UCPTtimeConfig.Byte[1] = 0 ==> <b>Display time</b> faded out	
UCPTtimeConfig.Byte[1] = 1 ==> <b>Display time</b> with seconds	
UCPTtimeConfig.Byte[1] = 2 ==> <b>Display time</b> without seconds	<b>Default</b>
UCPTtimeConfig.Byte[2] = 0 ==> <b>Display time mode</b> 24h	<b>Default</b>
UCPTtimeConfig.Byte[2] = 1 ==> <b>Display time mode</b> 12h	

### SCPTdriveTime

UCPT Index: 45, SNVT\_time\_sec

Function: This configuration property defines the updating interval of the displays in the main menu as well as the display period of a displayed message in case several messages are faded-in (see description of nviMessageX). (Preset value: 5,0 sec.)

### UCPTdisplayTime

UCPT Index: 16, SNVT\_time\_sec

Function: The configuration property defines the period of time of a display message respectively the up-date interval of the display. (Preset value: 5,0 sec.)

Node Object

**UCPTbacklightMin[2]**

UCPT Index: 48, SNVT\_lev\_cont

Function: In the stable state, i.e. after module reset or if no key actuation is made for a period longer than 15 sec. the value UCPTbacklightMin[0] determines the background illumination of the LCD. The value UCPTbacklightMax[1] determines the brightness of the inscription field.

**UCPTbacklightMax[2]**

UCPT Index: 47, SNVT\_lev\_cont

Function: If the stable state is left by actuating a key, the value UCPTbacklightMax[0] determines the brightness of the background illumination of the LCD. The value UCPTbacklightMax[1] determines the brightness of the inscription field.

**UCPTdisplConfig**

UCPT Index: 46, SNVT\_state

Function: By UCPTdisplConfig the display type and the values that shall be indicated in the main field can be configured. If several measuring values are shown, the display toggles between the measuring values in the time interval UCPTdisplayTime.

UCPTdisplConfig.Byte[0]	Function	Wert	Default
Byte[0].bit[0]	Rotation	0=Vertical 1=Horizontal	0
Byte[0].bit[1]	Background illumination LCD	0=black 1=white	0
Byte[0].bit[2]	Show temperature on/off	0 = hide 1 = show	1
Byte[0].bit[3]	Show humidity on/off	0 = hide 1 = show	0
Byte[0].bit[4]	Show decimal place temperature	0 = without a tenth 1 = with a tenth	1
Byte[0].bit[5]	Show decimal place rel. humidity	0 = without a tenth 1 = with a tenth	0
Byte[0].bit[6]	Activation of the device by actuating a key	0 = disabled 1 = enabled	0
Byte[0].bit[7]	Unit of the temperature	0 = °F 1 = °C	1
Byte[0].bit[8]	Show fan stage after reset	0 = Off 1 = On	0
Byte[0].bit[9]	Show room occupation after reset		1
Byte[0].bit[10]	Display menu line 1		0
Byte[0].bit[11]	Display menu line 2		0
Byte[0].bit[12]	Display menu line 3		0
Byte[0].bit[13]	Display menu line 4		0
Byte[0].bit[14]	Display menu line 5		0
Byte[0].bit[15]	Show fanstage in automatic mode		0

## Node Object

UCPTdisplConfig.Byte[1]	Function	value	Default
Byte[1].bit[0]	External value 1: resolution	0 = without decimal point 1 = with decimal point	0
Byte[1].bit[1]	External value 1: Show in display	0 = Off 1 = On	0
Byte[1].bit[2]	External value 2: resolution	0 = without decimal point 1 = with decimal point	0
Byte[1].bit[3]	External value 2: Show in display	0 = Off 1 = On	0
Byte[1].bit[4]	External value 3: Resolution	0 = without decimal point 1 = with decimal point	0
Byte[1].bit[5]	External value 3: Show in display	0 = Off 1 = On	0
Byte[1].bit[6]	External value 4: Resolution	0 = without decimal point 1 = with decimal point	0
Byte[1].bit[7]	External value 4: Show in display	0 = Off 1 = On	0
Byte[1].bit[8]	External value 5: resolution	0 = without decimal point 1 = with decimal point	0
Byte[1].bit[9]	External value 5: Show in display	0 = Off 1 = On	0
UCPTdisplConfig.Byte[2]	Function	value	Default
Byte[2].bit[0]	Set-point temperature regulator: SCC Resolution	0 = without decimal point 1 = with decimal point	1
Byte[2].bit[1]	Set-point temperature regulator: SCC Display after modification	0 = offset 1 = effectively	1
Byte[2].bit[2]	Set-point temperature regulator: SCC Show effective value in display	0 = Off 1 = On	0
Byte[2].bit[3]	Set-point temperature regulator: SCC Show offset value in display	0 = Off 1 = On	0
Byte[2].bit[4]	Set-point 1: OpenLoopSensor[0] Resolution	0 = without decimal point 1 = with decimal point	0
Byte[2].bit[5]	Set-point 1: OpenLoopSensor[0] Display after modification	0 = offset 1 = effectively	0
Byte[2].bit[6]	Set-point 1: OpenLoopSensor[0] Show effective value in display	0 = Off 1 = On	0
Byte[2].bit[7]	Set-point 1: OpenLoopSensor[0] Show offset value in display	0 = Off 1 = On	0
Byte[2].bit[8]	Set-point 2: OpenLoopSensor[1] resolution	0 = without decimal point 1 = with decimal point	0
Byte[2].bit[9]	Set-point 2: OpenLoopSensor[1] Display after modification	0 = offset 1 = effectively	0
Byte[2].bit[10]	Set-point 2: Show effective value in display	0 = Off 1 = On	0
Byte[2].bit[11]	Set-point 2: OpenLoopSensor[1] Show offset value in display	0 = Off 1 = On	0
Byte[2].bit[12]	Set-point 3: OpenLoopSensor[3] resolution	0 = without decimal point 1 = with decimal point	0
Byte[2].bit[13]	Set-point 3: OpenLoopSensor[3] Display after modification	0 = offset 1 = effectively	0

**Node Object**

Byte[2].bit[14]	Set-point 3: OpenLoopSensor[3] Show effective value in display	0 = Off 1 = On	0
Byte[2].bit[15]	Set-point 3: OpenLoopSensor[3] Show offset value in display	0 = Off 1 = On	0
<b>UCPTdisplConfig.Byte[3]</b>	<b>Function</b>	<b>value</b>	<b>Default</b>
Byte[3].bit[0]	Set-point 4: OpenLoopSensor[4] resolution	0 = without decimal point 1 = with decimal point	0
Byte[3].bit[1]	Set-point 4: OpenLoopSensor[4] Display after modification	0 = offset 1 = effectively	0
Byte[3].bit[2]	Set-point 4: OpenLoopSensor[4] Show effective value in display	0 = Off 1 = On	0
Byte[3].bit[3]	Set-point 4: OpenLoopSensor[4] Show offset value in display	0 = Off 1 = On	0
Byte[3].bit[4]	Set-point 5: OpenLoopSensor[5] resolution	0 = without decimal point 1 = with decimal point	0
Byte[3].bit[5]	Set-point 5: OpenLoopSensor[5] Display after modification	0 = offset 1 = effectively	0
Byte[3].bit[6]	Set-point 5: OpenLoopSensor[5] Show effective value in display	0 = Off 1 = On	0
Byte[3].bit[7]	Set-point 5: OpenLoopSensor[5] Show offset value in display	0 = Off 1 = On	0

**UCPTlongPressTime2**

UCPT Index: 78, typedef struct { SNVT\_time\_sec dimming; SNVT\_time\_sec sunblind;  
SNVT\_time\_sec scene; SNVT\_time\_sec universal; }

Function: The action for a long press can be defined with this configuration property, e.g. dimming, blind, scene and universal. (Preset value: 2.0;2.0;2.0;2.0)

**UCPTsunblindUP**

UCPT Index: 72, SNVT\_setting

Function: This configuration property defines, which SNVT\_setting value is transmitted while opening the blind.  
(Preset value: SET\_UP 100.0 0.0)

**UCPTsunblindDOWN**

UCPT Index: 73, SNVT\_setting

Function: This configuration property defines, which SNVT\_setting value  
Is transmitted while shutting the blinds..  
(Preset value: SET\_DOWN 100.0 0.0)

**UCPTsunblindSTOP**

UCPT Index: 74, SNVT\_setting

Function: This configuration property defines, which SNVT\_setting value is transmitted while stopping the blind.  
(Preset value: SET\_STOP 0.0 0.0)

**UCPTsunblindIDLE**

UCPT Index: 75, SNVT\_setting

Function: This configuration property defines, which SNVT\_setting value will be transmitted for the standby mode of the blind. UCPTsunblindIDLE is transmitted 500ms after stopping the blind, only if UCPTsunblindIDLE is unequal UCPTsunblindSTOP.  
(Preset value: SET\_NUL 0.0 0.0)

## Node Object

### UCPTnrOfSubmenus

UCPT Index: 79, unsigned short

Function: Selection, if a sub-menu shall be used. A sub-menu is only available in the S-version!!!

### UCPTshowSubMenuDuration

UCPT Index: 80, SNVT\_time\_sec

Function: This configuration property defines the display time of a submenu and the display menus.

### UCPTstring3

UCPT Index: 3, SNVT\_str\_asc

Function: By means of this property, the corresponding message text which shall be faded-in upon activation of the associated nviMessage in the display is configured.

### SCPTminSendTime

SCPT Index: 52, SNVT\_time\_sec

Function: This configuration parameter stipulates the sending interval of the output variables in the dimming mode. By means of input values = 0 the function is deactivated.  
(Preset value: 0.3 s)

### SCPTstepValue

SCPT Index: 92, SNVT\_lev\_cont

Function: This configuration parameter defines the step size of the variables nvoSwitch.value in the dimming mode.

### UCPTreverseDelay

UCPT Index: 14, SNVT\_count

Function: The configuration parameter defines the reverse delay with a rotation reverse of the blind motors in ms. Thus an alternate command e.g. from nvoSetting = SET\_UP to nvoSetting = SET\_DOWN is output with a delay. (Preset value: 500 ms)

### UCPTdiConfig[0]...[13]

UCPT Index: 44, typedef struct {unsigned short Byte[4]} UNVT\_str\_hex4

Function: This configuration parameter determines the key function.

UCPTdiConfig[0].Byte[x] configures **clip**

UCPTdiConfig[1].Byte[x] configures **key 1**

...

UCPTdiConfig[12].Byte[x] configures **key 14**

UCPTdiConfig[13].Byte[x] configures **key 15**

UCPTdiConfig[x].Byte[0] configures the function

Response via display UCPTdiConfig[x].Byte[0] > 0

No response via display UCPTdiConfig[x].Byte[0] = 0

UCPTdiConfig[x].Byte[1] configures set-point index and type

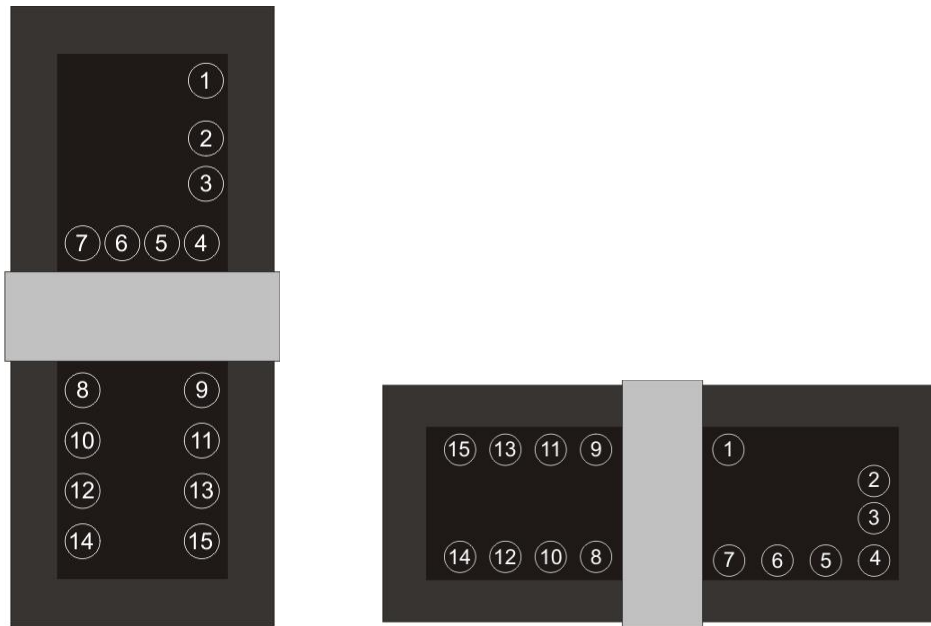
UCPTdiConfig[x].Byte[2] configures accompanying switch object

UCPTdiConfig[x].Byte[3] configures function of the key with no special function

**Notice: Keys 2 and 3 are assigned to the menus and cannot be field programmed!**



Node Object



Picture 3-1 Buttons **thanos L**

Instead of the lower direct keys, which are only available on **thanos L** / LQ, on **thanos S** / SQ up to 4 submenus can be configured.

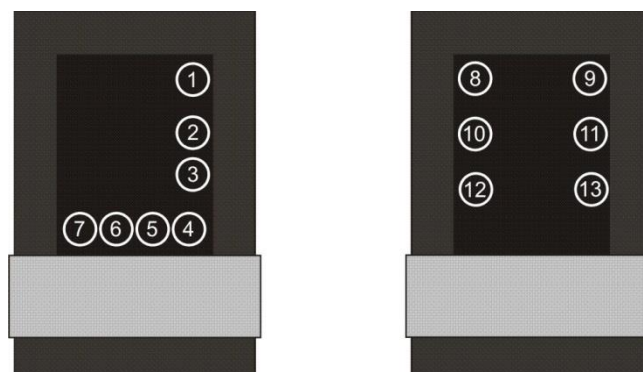
Up to 6 configurable keys for each submenu can be configured with the functions given below:

- Light on / off
- Light toggle
- Shutter / Blind up / down
- Universal on / off
- Universal toggle
- Occupancy toggle

To switch to the submenu, one of the keys 1, 4...7 has to be configured as „submenu right“.

The keys in the submenu are numbered consecutively (8 ... 13).

In addition, if a submenu should be used, the parameter "number of sub-menus" has to be set (range 0 ... 1).



Picture 8 Keys **thanos S**

### Menu area keys with symbol indication in the display

Selection of key functions via UCPTdiConfig[x].Byte[0]

Selection symbol and index UCPTdiConfig[x].Byte[1]

A set-point has to be connected with a symbol. For a better differentiation by the index the function can be marked.

Example:

Menu set-point 1 with symbol temperature and index 1

Byte[0] = 0x02

Byte[1] = 0x11

Selection Switch-Object UCPTdiConfig[x].Byte[2]

UCPTdiConfig, configuration of keys				
Taste 1...5 - function				
Byte[0] Function selection		Byte[1] Symbol and index selection		Byte[2] Switch-object selection
0x00	No special function	See: No special function (no indication in the display)		
0x01	Menu set-point temperature regulation SCC	0x1x Symbol Temperature	0xx0 Index 0 0xx1 Index 1 0xx2 Index 2 0xx3 Index 3 0xx4 Index 4 0xx5 Index 5 0xx6 Index 6 0xx7 Index 7 0xx8 Index 8 0xx9 Index 9	Not defined
0x02	Menu set-point 1 OpenLoopSensor[0]	0x1x Symbol Temperature 0x2x Symbol humidity 0x3x Symbol value		
0x03	Menu set-point 2 OpenLoopSensor[1]			
0x04	Menu set-point 3 OpenLoopSensor[2]			
0x05	Menu set-point 4 OpenLoopSensor[3]			
0x06	Menu set-point 5 OpenLoopSensor[4]			
0x07	Menu fan stage			
0x08	Menu Light			Selection Switch Object
0x09	Menu Light-Dimming			
0x0A	Menu blind			
0x0B	Menu Universal			
0x0C	Menu Presence			

### Keypad with user-defined channel assignment

Working with **thanos L** the keys on the keypad and working with **thanos S** the keys in the submenu can be configured with the following functions. Up to 10 channels (Light, blinds, ...) can be controlled.

Selection of key functions via UCPTdiConfig[x].Byte[0]

Selection Index UCPTdiConfig[x].Byte[1]

The function can be marked for a better differentiation by the index.

Example:

Light ON and index 1

Byte[0] = 0xA0

Byte[1] = 0x01

Selection Switch-Object UCPTdiConfig[x].Byte[2]

UCPTdiConfig, Configuration of Menu Keys			
Taste 0,6...13 - function			
Byte[0] function selection		Byte[1] Symbol and Index selection	Byte[2] Switch-Object selection
0x00	No special function	See: No special function (no indication in the display)	
0xA0	Light ON	0xx0 Index 0 0xx1 Index 1 0xx2 Index 2 0xx3 Index 3 0xx4 Index 4 0xx5 Index 5 0xx6 Index 6 0xx7 Index 7 0xx8 Index 8 0xx9 Index 9	Selection Switch Objekt
0xA1	Light OFF		
0xA2	Blind UP		
0xA3	Blind DOWN		
0xA4	Universal ON		
0xA5	Universal OFF		
0xA6	Light toggle		
0xA7	Universal toggle		
0xA8	Presence toggle		

### Keypad with fixed channel 0 assignment

Working with **thanos L** the keys on the keypad and working with **thanos S** the keys in the submenu can be configured with the following functions. All functions refer to channel 0.

UCPTdiConfig[x].Byte[0] = 0x00

Selection of key functions via UCPTdiConfig[x].Byte[3]

UCPTdiConfig, configuration of keys	
Byte[3]	key 1...14 - function
<b>No function</b>	
0x00	unoccupied

### Actuation function 0x01 – 0x05

UCPTdiConfig, configuration of keys		
Byte[3]	key 1...14 - function	SNVT-Type
<b>Actuation</b>		
0x01	Pressed/ unpressed	SNVT_switch SNVT_setting
0x02	Light toggle	SNVT_switch SNVT_setting
0x03	Light only ON	SNVT_switch SNVT_setting
0x04	Light only OFF	SNVT_switch SNVT_setting
0x05	Automatic	SNVT_switch

### 0x10 – 0x14 Dimming function

UCPTdiConfig, configuration of keys		
Byte[3]	Key 1...14 - function	SNVT-Type
<b>Dimming</b>		
0x10	Light toggle with dimming Switch-on value = Max-value	SNVT_switch
0x11	Light toggle with dimming Switch-on value = last On-value	SNVT_switch
0x12	Light only brighter with dimming Switch-on value = Max-value	SNVT_switch SNVT_setting
0x13	Light only brighter with dimming Switch-on value = last On-value	SNVT_switch SNVT_setting
0x14	Light only darker with dimming nvoMultiOut_1	SNVT_switch SNVT_setting

Short keystroke switches ON/OFF of the background illumination. With long keystroke light can be dimmed.

In toggle-mode the dimming-direction will be changed (brighter respectively darker) through a following keystroke.

0x20 – 0x21 blinds

UCPTdiConfig, configuration of keys		
Byte[3]	Taste 1...14 - function	SNVT-Type
Jalousie		
0x20	Blind UP	SNVT_setting
0x21	Blind DOWN	SNVT_setting

Short keystrokes result in stopping or adjusting the blinds. Long keystrokes activate the automatic opening or closing (0-100%).

0x30 – 0x31 shutters

UCPTdiConfig, configuration of keys		
Byte[3]	Taste 1...14 - function	SNVT-Type
Shutters		
0x30	Shutter UP	SNVT_setting
0x31	Shutter DOWN	SNVT_setting

The shutter opens or shuts as long as a button is pressed. Short keystrokes activate the automatic opening or shutting.

0x40 – 0x4F Scene-call

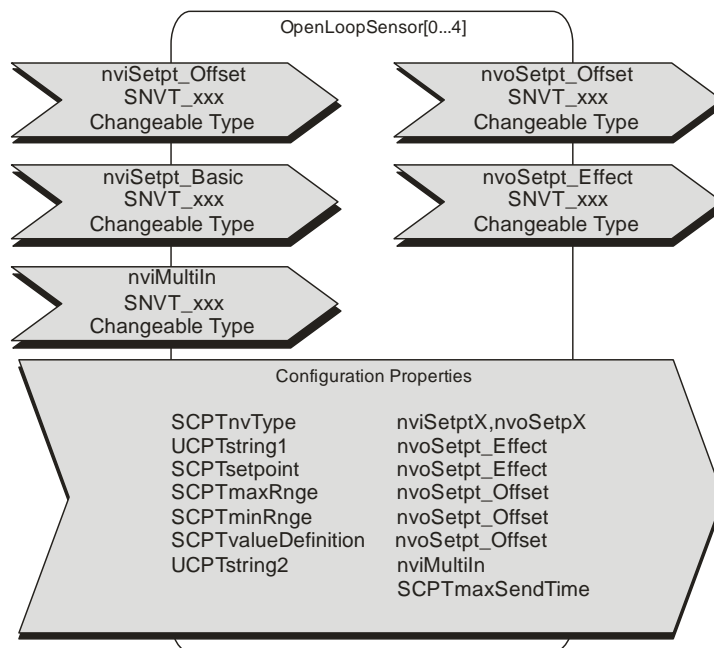
UCPTdiConfig, configuration of keys		
Byte[3]	Taste 1...14 - function	SNVT-Type
Scene-call		
0x40	Scene 0	SNVT_scene
0x41	Scene 1	SNVT_scene
...		
0x4F	Scene 15	SNVT_scene

Scenes 0-15 can be called up by short keystrokes.

Long keystrokes save a scene.

## 4 OpenLoop Sensor

Five identical objects to indicate 5 values and adjust 5 setpoints. Setpoints and values can be freely parameterized. Network variables can be changed.



### 4.1 Input variables

#### nviMultiIn

**SNVT Typ:** changeable\_type, i.e. the variable type can be configured via LON-installation tool e.g. the LonMaker. All 2 Byte-values can be shown. (default: SNVT\_switch)

**valid values:** SNVT\_switch, Index 95; SNVT\_lux, Index 79; SNVT\_lev\_percent, Index 81 ; SNVT\_ppm, Index 29; SNVT\_press, Index 30 ...

**Function:** The input variable is shown in the display depending on the configuration. Displayable values from -3276,7 to +3276,8.

**Parameter:** With the parameter UCPTString2[0] the unit (3 symbols) and with UCPTString2[1] a 4 symbol text have to be defined for the value to display.

#### nviSetpt\_Offset

**SNVT Typ:** changeable\_type, i.e. the variable type can be configured via LON-installation tool e.g. the LonMaker. All 2 Byte-values can be shown. (default: SNVT\_switch)

**Valid values:** SNVT\_switch, Index 95; SNVT\_lux, Index 79; SNVT\_lev\_percent, Index 81 ; SNVT\_ppm, Index 29; SNVT\_press, Index 30 ...

**Function:** The input variable to overwrite the set-point offset. Displayable values from -3276,7 to +3276,8..

#### nviSetpt\_Basic

**SNVT Typ:** changeable\_type, i.e. the variable type can be configured via LON-installation tool e.g. the LonMaker. All 2 Byte-values can be shown. (default: SNVT\_switch)

**Valid values:** SNVT\_switch, Index 95; SNVT\_lux, Index 79; SNVT\_lev\_percent, Index 81 ; SNVT\_ppm, Index 29; SNVT\_press, Index 30 ...

**Function:** The input variable to overwrite the standard set-point. Displayable values from -3276,7 to +3276,8..

## 4.2 Output Variables

### nvoSetpt\_Offset

SNVT Typ: changeable\_type, i.e. the variable type can be configured via LON-installation tool e.g. the LonMaker.  
All 2 Byte-values can be shown.  
(default: SNVT\_switch)

Valid values: SNVT\_switch, Index 95; SNVT\_lux, Index 79; SNVT\_lev\_percent, Index 81 ; SNVT\_ppm, Index 29;  
SNVT\_press, Index 30 ...

Function: Output variable for the set-point offset.  
Displayable values from -3276,7 to +3276,8..

Parameter: SCPTminRnge, SCPTmaxRnge minimal and maximal set-point offset.  
SCPTvalueDefinition Value for the set-point offset, due to which set-point it should be changed.

### nvoSetpt\_Effect

SNVT Typ: changeable\_type, i.e. the variable type can be configured via LON-installation tool e.g. the LonMaker.  
All 2 Byte-values can be shown.  
(default: SNVT\_switch)

Valid values: SNVT\_switch, Index 95; SNVT\_lux, Index 79; SNVT\_lev\_percent, Index 81 ; SNVT\_ppm, Index 29;  
SNVT\_press, Index 30 ...

Function: Output variable for the effective set-point.  
Displayable values from -3276,7 to +3276,8..

Parameter: With the parameter UCPTString2[0] the unit (3 symbols) and with UCPTString2[1] a 4 symbol text  
have to be defined for the value to display.

## 4.3 Configuration Property Open Loop Sensor

### SCPTnvType

SCPT Index: 254, SNVT\_nv\_type

Function: There is a SCPTnvType for nvoMultiOut\_1 and nvoMultiOut\_2 available each. This configuration property specifies the type of the network variable nvoMultiOut\_1 respectively nvoMultiOut\_2. If SCPTnvType is not automatically adapted to the new variable type of nvoMultiOut\_1 / nvoMultiOut\_2 by the installation tool the following settings must be made.

nvoMultiOut = SNVT\_switch

==> SCPTnvType = PID 0:0:0:0:0:0:0:0, Scope 0, Index 95, NVT\_CAT\_STRUCT, 2 bytes, A=1, B=0, C=0

nvoMultiOut = SNVT\_setting

==> SCPTnvType = PID 0:0:0:0:0:0:0:0, Scope 0, Index 117, NVT\_CAT\_STRUCT, 4 bytes, A=1, B=0, C=0

nvoMultiOut = SNVT\_lev\_percent

==> SCPTnvType = PID 0:0:0:0:0:0:0:0, Scope 0, Index 81, NVT\_CAT\_SIGNED\_LONG, 2 bytes, A=5, B=-3, C=0

nvoMultiOut = SNVT\_lux

==> SCPTnvType = PID 0:0:0:0:0:0:0:0, Scope 0, Index 79, NVT\_CAT\_UNSIGNED\_LONG, 2 bytes, A=1, B=0, C=0

nvoMultiOut = SNVT\_occupancy

==> SCPTnvType = PID 0:0:0:0:0:0:0:0, Scope 0, Index 109, NVT\_CAT\_ENUM, 1 bytes, A=1, B=0, C=0

nvoMultiOut = SNVT\_scene

==> SCPTnvType = PID 0:0:0:0:0:0:0:0, Scope 0, Index 115, NVT\_CAT\_STRUCT, 2 bytes, A=1, B=0, C=0

nvoMultiOut = SNVT\_ppm

==> SCPTnvType = PID 0:0:0:0:0:0:0:0, Scope 0, Index 29, NVT\_CAT\_UNSIGNED\_LONG, 2 bytes, A=1, B=0, C=0

### UCPTstring1

Index: 1, SNVT\_str\_asc

Function: With this property the text for the unit (max. 3 ASCII signs) and the description (max. 4 ASCII signs) of the set point are set. These texts are displayed when the set point is faded-in.

**UCPTstring2**

Index: 2, SNVT\_str\_asc

Function: With this property the text for the unit (max. 3 ASCII signs) and the description (max. 4 ASCII signs) of the external value set point are set. These texts are displayed when the external value is faded-in.

**SCPTsetpoint**

Index: 213, signed long

Function: Basic setpoint after reset

**SCPTmaxRange**

Index: 20, signed long

Function: The maximum limit of the value of the primary output network variable for the object

**SCPTminRange**

Index: 23, signed long

Function: The minimum limit of the value of the primary output network variable for the object

**SCPTvalueDefinition**

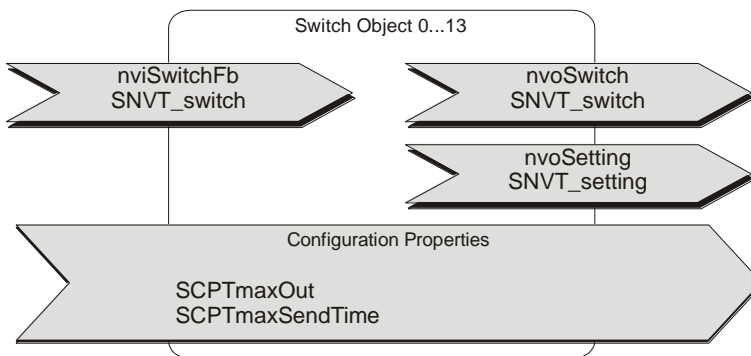
Index: 23, signed long

Function: This configuration parameter defines the step size of the nvoSetpt\_Offset variable.



## Switch Object

### 5 Switch Object



14 identical switch objects for light and blind control. Both, the allocation of the keys to the individual objects and the determination of the functions are made by the configuration parameters UCPTdiConfig[0]...[13] in the Node Object.

Possible functions are light control ON/OFF, light control ON/OFF with dimming or blinds UP/DOWN with automatic run.

#### 5.1 Input Variable Switch Object:

##### nviSwitchFb

SNVT Typ: SNVT\_switch, Index 95

Function: Input Variable for the current state of the nvoSwitch controlled illumination groups. **This variable is only active when a menu button is configured with “menu light” or “menu universal”, or a key on the keypad with “light toggle”, universal toggle” or dimming function!**

**For toggle and dimming function a feedback with the current state of the controlled illumination group is mandatory!**

#### 5.2 Output Variable Switch Object:

##### nvoSwitch / nvoSetting

SNVT Typ: SNVT\_switch, Index 95 / SNVT\_setting, Index 119

Function: Depending on the configuration UCPTdiConfig the output variables send the current switch status of the key (active/inactive) or values for manual light control.

##### Switch/ Key

###### Key pressed/ not pressed

SNVT Type: SNVT\_switch

Key pressed      nvoSwitch.value      = SCPTmaxOut

nvoSwitch.state      = 1

Key not pressed      nvoSwitch.value      = 0

nvoSwitch.state      = 0

SNVT Type: SNVT\_setting

Key pressed      nvoSetting.function      = SET\_ON

nvoSetting.setting      = SCPTmaxOut

Key not pressed      nvoSetting.function      = SET\_OFF

nvoSetting.setting      = 0

###### Lighting Toggle

Each key actuation results in a toggling of the lighting, i.e. between ON and OFF

SNVT Type: SNVT\_switch

Lighting ON      nvoSwitch.value      = SCPTmaxOut

nvoSwitch.state      = 1

Lighting OFF      nvoSwitch.value      = 0

nvoSwitch.state      = 0

SNVT Type: SNVT\_setting

Lighting ON      nvoSetting.function      = SET\_ON

nvoSetting.setting      = SCPTmaxOut

## Switch Object

Lighting OFF	nvoSetting.function	= SET_OFF;
	nvoSetting.setting	= 0;

### Lighting ON

Each key actuation results in a toggling of the lighting

SNVT Type: SNVT_switch		
Lighting ON	nvoSwitch.value	= SCPTmaxOut
	nvoSwitch.state	= 1
SNVT Type: SNVT_setting		
Lighting ON	nvoSetting.function	= SET_ON
	nvoSetting.setting	= SCPTmaxOut

### Lighting OFF

Each key actuation results in a switching-off of the lighting

SNVT Type: SNVT_switch		
Lighting OFF	nvoSwitch.value	= 0
	nvoSwitch.state	= 0
SNVT Type: SNVT_setting		
Lighting OFF	nvoSetting.function	= SET_OFF
	nvoSetting.setting	= 0

### Automatic

The actuation of an “Automatic-Key“ switches the variable nvoSwitch to the value 0,0 -1. Thus, e.g. a light controller can be reset in the automatic mode after external override.

SNVT Type: SNVT\_switch

## Dimming

### Lighting: Toggle by Dimming, Switch-On Value = max. Value

Short key acutations result in a toggling of the current lighting status, whereas the .value –turn-on value always is SCPTmaxOut. By longer key actuations the dimming function is activated, i.e. based on the current lighting status, the .value-value of the switch variables is raised or lowered in precent steps of UCPTstepValue as long as the key is pressed. A renewed long time key actuation results in a reversal of the dimming direction.

SNVT Type: SNVT_switch		
Lighting on maximum value	nvoSwitch.value	= SCPTmaxOut
	nvoSwitch.state	= 1
Lighting on 50%	nvoSwitch.value	= 50,0
	nvoSwitch.state	= 1
Lighting OFF	nvoSwitch.value	= 0
	nvoSwitch.state	= 0
SNVT Typ: SNVT_setting		
Beleuchtung auf Maximalwert	nvoSetting.function	= SET_ON
	nvoSetting.setting	= SCPTmaxOut
Beleuchtung Heller-Dimmen	nvoSetting.function	= SET_UP
	nvoSetting.setting	= SCPTstepValue
Beleuchtung Dunkler-Dimmen	nvoSetting.function	= SET_DOWN
	nvoSetting.setting	= SCPTstepValue
Beleuchtung AUS	nvoSetting.function	= SET_OFF
	nvoSwitch.setting	= 0

## Switch Object

### Lighting: Toggle by Dimming, Switch-ON Value = Last Switch-ON Value

Function as with 50<sub>hex</sub>, but with the difference, that not the value SCPTmaxOut but the last turn-on value is taken over. The smallest turn-on value is limited to 20%.

### Lighting ON by Brighter-Dimming, Switch-ON Value = max. Value

If the lighting is switched-off, a key actuation results in an immediate switching-on of the lighting. By longer key actuations the function “dim brighter“ is activated, i.e. based on the current light status the .value – value of the switch variable is reduced in percent steps of UCPTstepValue as long as the maximum value SCPTmaxOut is reached. The sending interval in the mode dimming is adjusted by SCPTminSendTime and is preset to approx. 300ms.

SNVT Type: SNVT\_switch

Switching-on of lighting	nvoSwitch.value	= SCPTmaxOut
	nvoSwitch.state	= 1
Brighter dimming of lighting	nvoSwitch.value	= last value + UCPTstepValue
	nvoSwitch.state	= 1

SNVT Type: SNVT\_setting

Switching-on of lighting	nvoSetting.function	= SET_ON
	nvoSetting.setting	= SCPTmaxOut
Brighter dimming of lighting	nvoSetting.function	= SET_UP
	nvoSetting.setting	= UCPTstepValue

### Lighting ON by Brighter Dimming, Switch-ON Value = last ON-value

Function as with 52hex but with the difference, that not the value SCPTmaxOut is taken over when switching-on the light, but the last turn-on value. The smallest turn-on value is limited to 20%.

### Lighting OFF by Darker Dimming

If the lighting is turned-on, a short key actuation leads to an immediate switching-off of the lighting. By longer key actuations the function “dim darker“ is activated, i.e. based on the current lighting status the .value –value of the switch variables is reduced in percent steps of UCPTstepValue as long as the value 0 is reached. The sending interval in the mode dimming is adjusted by SCPTminSendTime and amounts to approx. 300ms preset.

SNVT Type: SNVT\_switch

Switching-off of lighting	nvoSwitch.value	= 0
	nvoSwitch.state	= 0
Darker dimming of lighting	nvoSwitch.value	= last value - UCPTstepValue
	nvoSwitch.state	= 1

SNVT Type: SNVT\_setting

Switching-off of lightning	nvoSetting.function	= SET_OFF
	nvoSetting.setting	= 0
Darker dimming of lightning	nvoSetting.function	= SET_DOWN
	nvoSetting.setting	= SCPTstepValue

## Switch Object

### Blind

#### Blind UP

In the configuration mode "blind UP" only the nvoSetting variables are changed and sent. Short key actuations are used for a fine adjustment of the lamellas. A long key actuation starts the automatic run and drives the blind continuously in the direction open for the time SCPTdriveTime. The automatic run can be stopped by a renewed key actuation.

SNVT Type: SNVT\_setting

Open blind	nvoSetting.function	= UCPTsunblindUP
Stop blind	nvoSetting.function	= UCPTsunblindSTOP

With a delay of 500ms the command UCPTsunblindIDLE for idle mode is sent after the command UCPTsunblindSTOP if UCPTsunblindIDLE is unequal to UCPTsunblindSTOP.

#### Blind DOWN

In the configuration mode "blind DOWN" only the nvoSetting variables are changed and sent. Short key actuations are for the fine adjustment of the lamellas. A long key actuation starts the automatic run and drives the blind for the time SCPTdriveTime continuously into the direction close. The automatic run can be stopped by a renewed key actuation.

SNVT Type: SNVT\_setting

Close blind	nvoSetting.function	= UCPTsunblindDOWN
stop blind	nvoSetting.function	= UCPTsunblindSTOP

With a delay of 500ms the command UCPTsunblindIDLE for idle mode is sent after the command UCPTsunblindSTOP if UCPTsunblindIDLE is unequal to UCPTsunblindSTOP.

### Shutter

#### Shutter UP

In the configuration mode "Shutter UP" only the nvoSetting variables are changed and sent. Short key actuation starts the automatic run and drives the shutter continuously in the direction open for the time SCPTdriveTime. The automatic run can be stopped by a renewed key actuation. By a long key actuation the position of the shutter can be individually adjusted.

SNVT Type: SNVT\_setting

Open blind	nvoSetting.function	= UCPTsunblindUP
Stop blind	nvoSetting.function	= UCPTsunblindSTOP

With a delay of 500ms the command UCPTsunblindIDLE for idle mode is sent after the command UCPTsunblindSTOP if UCPTsunblindIDLE is unequal to UCPTsunblindSTOP.

#### Shutter down

In the configuration mode "shutter DOWN" only the nvoSetting variables are changed and sent. Short key actuation starts the automatic run and drives the shutter continuously into the direction close for the time SCPTdriveTime. The automatic run can be stopped by a renewed key actuation. By a long key actuation the position of the shutter can be adjusted individually.

SNVT Type: SNVT\_setting

Close shutter	nvoSetting.function	= UCPTsunblindDOWN
Stop shutter	nvoSetting.function	= UCPTsunblindSTOP

With a delay of 500ms the command UCPTsunblindIDLE for idle mode is sent after the command UCPTsunblindSTOP if UCPTsunblindIDLE is unequal to UCPTsunblindSTOP.

### 5.3 Configuration Parameters Switch Object:

#### SCPTmaxOut

SCPT Index: 93, SNVT\_lev\_cont

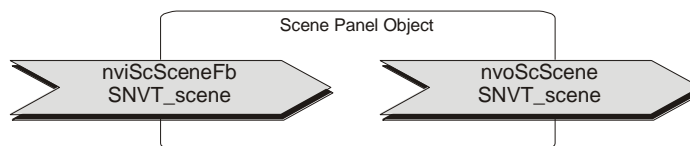
Function: Configuration property to restrict the Value of the Output Variable nvoSwitch.value.  
(Preset value: 100 %)

#### SCPTmaxSendTime

SCPT Index: 49, SNVT\_time\_sec

Function: Heartbeat function. This configuration property stimulates the interval time after which the output variables are sent. By means of input values =0, the heartbeat function is deactivated.  
(Preset value: 0 s)

## 6 Scene Panel-Objekt



### 6.1 Eingangsvariablen Scene Panel:

#### nviScSceneFb

SNVT Typ: SNVT\_scene, Index 115

Funktion: Input variable for current illumination scene. If a new input value is received the key an led configuration will be checked. If configured, leds indicates the current scene.

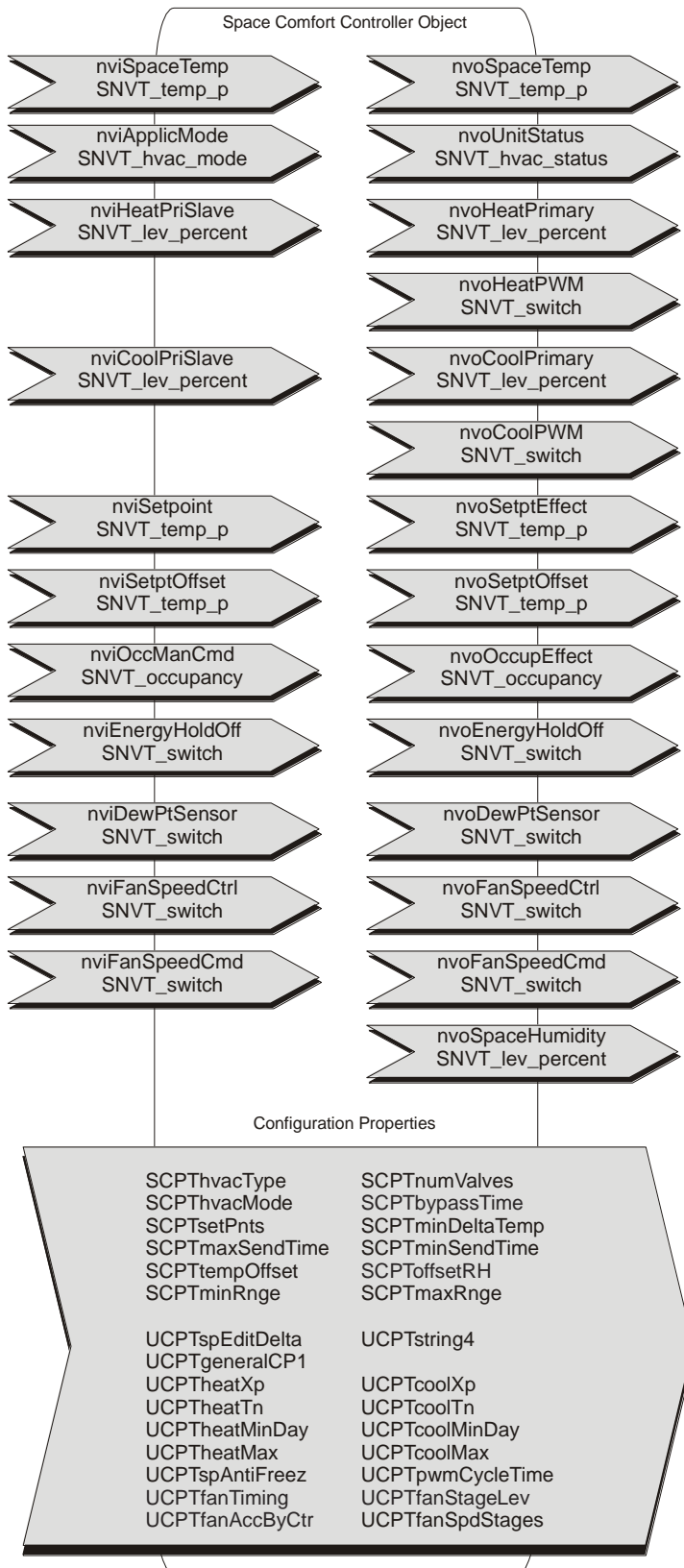
### 6.2 Ausgangsvariablen Scene Panel:

#### nvoScScene

SNVT Typ: SNVT\_scene, Index 115

Funktion: Output variable for control of a scene controller. The scene numbers 0-15 can be allocated to the key. With short key actuations the scene is called by SC\_RECALL. With long key actuations, the scene is learned-in again by SC\_LEARN. The output is made to nvoScScene.

## 7 Space Comfort Controller



The Object includes the functions temperature measuring and PID-control for heating and cooling. Output of the control variables is made via the network variables. The scanning time for the calculation of the control variables corresponds to the sending interval (SCPTmaxSendTime) of the output variables.

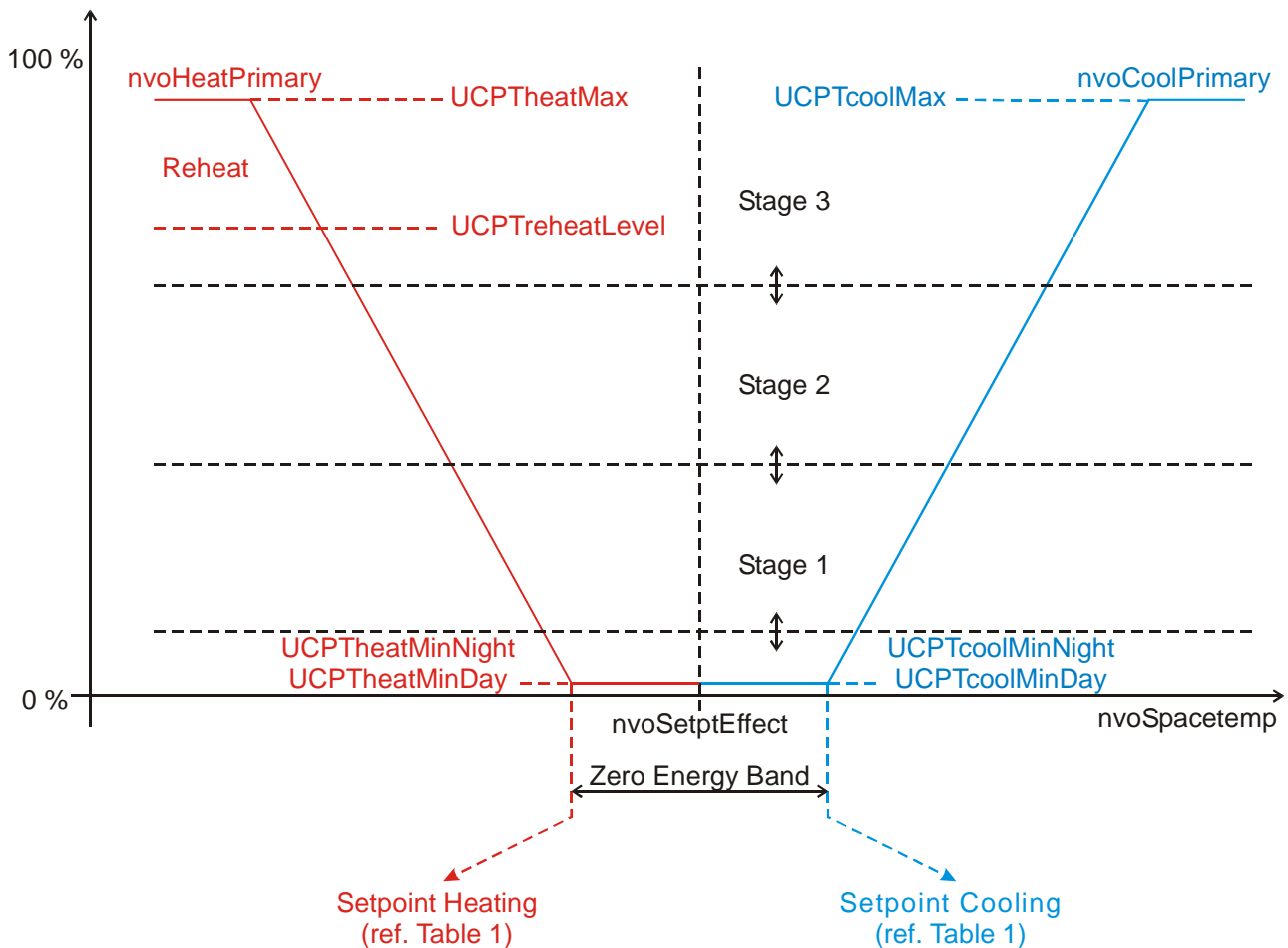
**Set point adjustment:** The set point temperature can be increased/ lowered in the range of UCPTspOffsetRng via the menu keys. Output of the offset value is made by nvoSetptOffset.

The effective set point (basic set point) nvoSetptEffect is calculated in reliance on the input variables to the room occupancy (nviOccManCmd and nviOccSensor) of the set point defaults SCPTsetPnts respectively nviSetpoint and the set point offset.

**Temperature Control:** The basic set point used by the control algorithm is determined by SCPTsetPnts respectively nvSetpoint. The neutral (energy-free) zone around the basic set point adapts itself automatically to the current room occupancy and is paramaterizable via SCPTsetPnts. The control variables of the PI-controller for heating and cooling are output by the variables of type SNVT\_lev\_percent.

The control properties for the proportional range and reset time can be individually adapted to the room conditions. The monitoring of window contacts and dew point detector are made by the input variables nviEnergyHoldOff and nviDewPtSensor.

### Function diagram PI-Controller Heating/Cooling:



## 7.1 Input Variables Space Comfort Controller Object

### nviSpaceTemp

SNVT Type: SNVT\_temp\_p, Index 105

Function: Input variable for connection of an external LON temperature sensor. The external value is taken over, if the initialisation value 0x7FFF (=327,67 °C) was changed by a NV-Update after reset. As long as the initialisation value is not changed after reset, the internal temperature sensor remains active

### nviApplicMode

SNVT Type: SNVT\_hvac\_mode, Index 108

Function: Input variable for selection of the controller operating mode.  
 HVAC\_AUTO ==> automatic toggling between heating and cooling  
 HVAC\_HEAT ==> only heating  
 HVAC\_COOL ==> only cooling  
 HVAC\_OFF ==> control switched-off  
 The initialization status after reset is determined by the configuration property SCPT hvacMode.

**nviCoolPriSlave**

SNVT Type: SNVT\_lev\_percent, Index 81

Function: Control variable for network variables nvoCoolPrimary and nvoCoolPWM.

nviCoolPriSlave = 0x7FFF (163,835 %) ==&gt; internal controller cooling ON (initialisation value after reset).

nviCoolPriSlave = 0...100 % ==&gt; internal controller cooling OFF

==&gt; nviCoolPriSlave determines the output variables.

The symbol COOLING is driven in dependence of the value of this input variable, when the internal controller is deactivated (SCPThvacMode=HVAC\_OFF und nviApplicMode = HVAC\_OFF). It is fade-in when the value of the input variable is between >0% and <= 100%.

**nviHeatPriSlave**

SNVT Type: SNVT\_lev\_percent, Index 81

Function: Control variable for network variable nvoHeatPrimary and nvoHeatPWM.

nviHeatPriSlave = 0x7FFF (163,835 %) ==&gt; internal controller heating ON (initialisation value after reset)

nviHeatPriSlave = 0 ... 100 % ==&gt; internal controller heating OFF

==&gt; nviHeatPriSlave determines the output quantities

The symbol HEATING is driven in dependence of the value of this input variable, when the internal controller is deactivated (SCPThvacMode=HVAC\_OFF und nviApplicMode = HVAC\_OFF). It is fade-in when the value of the input variable is between >0% and <= 100%.

**nviOccManCmd**

SNVT Typ: SNVT\_occupancy, Index 109

Function: Input variable for default of the room occupancy. The current room occupancy determines the values of the control parameter „effective set point“ and „neutral zone“ and thus the set points for heating and cooling (see table 1). Initialisation value for both variables: OC\_NUL

nviOccManCmd: default with GLT via: OC\_OCCUPIED, OC\_STANDBY, OC\_UNOCCUPIED

Via the configuration property *UCPTgeneralCPI* the behaviour of the device can be determined if nviOccManCmd = OC\_UNOCCUPIED is switched.



nviOccManCmd	Internal room occupation	□ □ □	room occupancy nvoOccupEffect	Setpoint Heat nvoSetptEffect (nvoUnitStatus.mode = HVAC_HEAT)	Setpoint Cool nvoSetptEffect (nvoUnitStatus.mode = HVAC_COOL)
OC_NUL	OC_NUL	>>>	OCCUPIED	$\begin{aligned} & (\text{SCPTsetPnts.occupied\_cool} + \\ & \text{SCPTsetPnts.occupied\_heat}) / 2 \\ & + \\ & \text{nviSetptOffset} \\ & \text{or} \\ & \text{nviSetpoint} + \text{nviSetptOffset} \end{aligned}$	$\begin{aligned} & (\text{SCPTsetPnts.occupied\_cool} + \\ & \text{SCPTsetPnts.occupied\_heat}) / 2 \\ & + \\ & \text{nviSetptOffset} \\ & \text{or} \\ & \text{nviSetpoint} + \text{nviSetptOffset} \end{aligned}$
OC_OCCUPIED	****	>>>			
****	OC_OCCUPIED	>>>			
OC_STANDBY	OC_NUL OC_UNOCCUPIED	>>>	STANDBY	$\begin{aligned} & (\text{SCPTsetPnts.occupied\_cool} + \\ & \text{SCPTsetPnts.occupied\_heat}) / 2 \\ & - \\ & (\text{SCPTsetPnts.occupied\_heat} - \\ & \text{SCPTsetPnts.standby\_heat}) \\ & + \\ & \text{nviSetptOffset} \\ & \text{or} \\ & \text{nviSetpoint} - \\ & (\text{SCPTsetPnts.occupied\_heat} - \\ & \text{SCPTsetPnts.standby\_heat}) \\ & + \\ & \text{nviSetptOffset} \end{aligned}$	$\begin{aligned} & (\text{SCPTsetPnts.occupied\_cool} + \\ & \text{SCPTsetPnts.occupied\_heat}) / 2 \\ & + \\ & (\text{SCPTsetPnts.occupied\_heat} - \\ & \text{SCPTsetPnts.standby\_heat}) \\ & + \\ & \text{nviSetptOffset} \\ & \text{or} \\ & \text{nviSetpoint} + \\ & (\text{SCPTsetPnts.occupied\_heat} - \\ & \text{SCPTsetPnts.standby\_heat}) \\ & + \\ & \text{nviSetptOffset} \end{aligned}$
OC_UNOCCUPIED	OC_NUL OC_UNOCCUPIED	>>>	UNOCCUPIED	$\begin{aligned} & (\text{SCPTsetPnts.occupied\_cool} + \\ & \text{SCPTsetPnts.occupied\_heat}) / 2 \\ & - \\ & (\text{SCPTsetPnts.occupied\_heat} - \\ & \text{SCPTsetPnts.unoccupied\_heat}) \\ & + \\ & \text{nviSetptOffset} \\ & \text{or} \\ & \text{nviSetpoint} - \\ & (\text{SCPTsetPnts.occupied\_heat} - \\ & \text{SCPTsetPnts.unoccupied\_heat}) \\ & + \\ & \text{nviSetptOffset} \end{aligned}$	$\begin{aligned} & (\text{SCPTsetPnts.occupied\_cool} + \\ & \text{SCPTsetPnts.occupied\_heat}) / 2 \\ & + \\ & (\text{SCPTsetPnts.occupied\_heat} - \\ & \text{SCPTsetPnts.unoccupied\_heat}) \\ & + \\ & \text{nviSetptOffset} \\ & \text{or} \\ & \text{nviSetpoint} + \\ & (\text{SCPTsetPnts.occupied\_heat} - \\ & \text{SCPTsetPnts.unoccupied\_heat}) \\ & + \\ & \text{nviSetptOffset} \end{aligned}$

**Table 1: Control property depending on the room occupancy**

**nviFanSpeedCmd**

SNVT Typ: SNVT\_switch, Index 95

Function: Input variable for external default of the displayed and with *nvoFanSpeed* transmitted fan-stage setting. The value range is equal to the Output Variable *nvoFanSpeed*.**nviFanSpeedCtrl**

SNVT Typ: SNVT\_switch, Index 95

Function: The input variable *nviFanSpeedCtrl* enables the display of the current fan speed if the externally connected fan coil controller selects the fan speed autonomously by *nviFanSpeedCmd* 0,0 -1. Therefore, the output variable of the controller (e.g. *nvoFanSpeedCmd*) must be connected with the input variable *nviFanSpeedCtrl*.Example: *nviFanSpeedCmd* = 0,0 -1 (Automatic mode) and *nviFanSpeedCtrl* = 33,0 1 => Display shows Automatic Stage 1 (assumed that the parameter "Display fan stage in auto mode" is activated). The output variable *nvoFanSpeedCtrl* outputs 33,0 1.**nviSetpoint**

SNVT Type: SNVT\_temp\_p, Index 105

Function: Input variable for default of the basic set point temperature . It is obligatory necessary to bind these network variables with a higher node. If no update is made for *nviSetpoint*, the initialisation value 0x7FFF (=327,67°C) is maintained and the values of the configuration properties SCPTsetPnts are used for the calculation of the effective set point (basic set point + Offset). If *nviSetpoint* receives an update with a valid set point, the effective set point is calculated by the value of the input variables.**nviSetptOffset**

SNVT Type: SNVT\_temp\_p, Index 105

Function: Input variable for default of an offset value for the offset of the basic set point temperature in the mode OCCUPIED or STANDBY (see table 1).

**nviEnergyHoldOff**

SNVT Type: SNVT\_switch, Index 95

Function: Input variable of e.g. window or door contact for activation of the energy-saving function. By *nviEnergyHoldOff* = 100.0 1 the function is activated and the control variables heating/cooling are reset to their minimum values. With an active energy-saving function the antifreeze function is switched on (see UCPTspAntiFreez). After deactivation of the energy-hold-off function the temperature control is restarted.**nviDewPtSensor**

SNVT Type: SNVT\_switch, Index 95

Function: Input variable for evaluation of a dew point detector in the operating mode cooling. By *nviDewPtSensor* = 100.0 1 the control variable cooling is reset to its minimum value. After deactivation of this function, the temperature control is started anew.

## 7.2 Output Variables Space Comfort Controller Object

**nvoSpaceTemp**

SNVT Type: SNVT\_temp\_p, Index 105

Function: Output variable for measured temperature value. Measuring range 0 - 50°C, resolution 1/100 °C. Data output is made depending of SCPTmaxSendTime, and approx. 1,5s - 4,0s after reset.

### nvoUnitStatus

SNVT Type: SNVT\_hvac\_status, Index 112

Function: Output variable for operating mode and the control variables heating/cooling of the controller.

.mode	=	HVAC_HEAT	==> heating
		HVAC_COOL	==> cooling
		HVAC_OFF	==> controller switched-off
.heat_output_primary	0...100 %		==> control variable heating
.cool_output_primary	0...100 %		==> control variable cooling

### nvoHeatPrimary

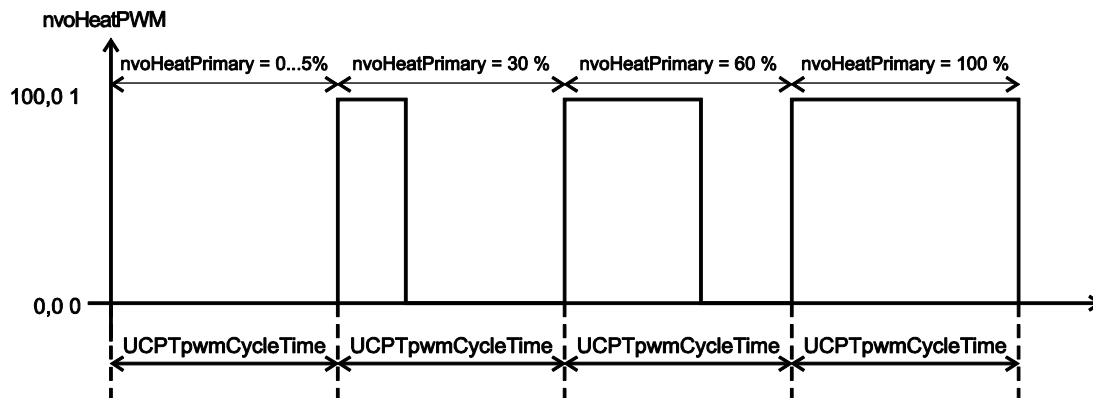
SNVT Type: SNVT\_lev\_percent, Index 81

Function: Output variable including the control variable of the PID-controller for heating to control a continuous actuator. Data output is made depending on SCPTmaxSendTime and 1,5s- 4s after reset.

### nvoHeatPWM

SNVT Type: SNVT\_switch, Index 95

Function: Output variable including the control variable of the PID-controller for heating to control a thermic two-point actuator pulse width modulated. Data output is made immediately with a waiting switch command and otherwise depending on SCPTmaxSendTime and 1,5s- 4s after reset.



### nvoCoolPrimary

SNVT Type: SNVT\_lev\_percent, Index 81

Function: Output variable including the control variable of the PID-controller for cooling. Data output is made analogue to nvoHeatPrimary.

### nvoCoolPWM

SNVT Type: SNVT\_switch, Index 95

Function: Output variable including the control variable of the PID-controller for heating for a pulse width modulated control of a thermic two-point actuator. Data output is made analogue to nvoHeatPWM.

### nvoSetptEffect

SNVT Type: SNVT\_temp\_p, Index 105

Function: The output variable sends the set point used by the control algorithm. Output is depending on the operating mode of the controller:

nvoUnitStatus.mode = HVAC_HEAT	==> nvoSetptEffect = set point heating
nvoUnitStatus.mode = HVAC_Cool	==> nvoSetptEffect = set point cooling

The effective set point is calculated depending on *nviSetpoint*, *nviOccManCmd*, *nviOccSensor*, *SCPTsetPnts* and *nviSetptOffset* (see table 1). Data output is made depending on *SCPTmaxSendTime*, upon value change and 1,5s- 4s after reset.

### nvoSetptOffset

SNVT Type: SNVT\_temp\_p, Index 105

Function: Output variable for set point correction, which can be prescribed by nviSetptOffset. Data output is made analogue to nvoSetptEffect.

### nvoOccupEffect

SNVT Type: SNVT\_occupancy, Index 109

Function: Output variable for effective room occupancy (see table 1). Data output is made depending on SCPTmaxSendTime, upon value change and 1,5s- 4s after reset.

### nvoEnergyHoldOff

SNVT Type: SNVT\_switch, Index 95

Function: Output variable for status message of the energy-saving function.

nvoEnergyHoldOff = 0.0 0 ==> window contact inactive

nvoEnergyHoldOff = 100.0 1 ==> window contact active

Data output is made depending on SCPTmaxSendTime, upon value change and 1,5s- 4s after reset.

### nvoDewPtSensor

SNVT Type: SNVT\_switch, Index 95

Function: Output variable for status message of the dew point detector.

nvoDewPtSensor = 0.0 0 ==> dew point detector inactive

nvoDewPtSensor = 100.0 1 ==> dew point

Data output is made depending on SCPTmaxSendTime, upon and 1,5s- 4s after reset.

### nvoFanSpeedCtrl

SNVT Type: SNVT\_switch, Index 95

Function: Output variable for adjusted fan speed. The number of the fan speed stages is adjusted by the configuration property UCPTfanSpdStages. The indication in the display shows the currently adjusted fan speed and whether the controller is in the automatic mode.

UCPTfanSpdStages = 1

Lüfterstufe	nvoFanSpeedCtrl .value	nvoFanSpeedCtrl .state
0	0 %	0
1	100 %	1

UCPTfanSpdStages = 2

Lüfterstufe	nvoFanSpeedCtrl .value	nvoFanSpeedCtrl .state
0	0 %	0
1	50 %	1
2	100 %	1

UCPTfanSpdStages = 3

Lüfterstufe	nvoFanSpeedCtrl .value	nvoFanSpeedCtrl .state
0	0 %	0
1	33,0 %	1
2	66,5 %	1
3	100 %	1

### nvoFanSpeedCmd

SNVT Typ: SNVT\_switch, Index 95

Function: Output Variable for the set fan stage and for combining several devices. The number of devices is adjustable with the configuration property UCPTfanSpdStages. The indication in the display shows the currently adjusted fan speed and whether the controller is in the automatic mode.

UCPTfanSpdStages = 1

Lüfterstufe	nvoFanSpeedCtrl .value	nvoFanSpeedCtrl .state
0	0 %	0
1	100 %	1

UCPTfanSpdStages = 2

Lüfterstufe	nvoFanSpeedCtrl .value	nvoFanSpeedCtrl .state
0	0 %	0
1	50 %	1
2	100 %	1

UCPTfanSpdStages = 3

Lüfterstufe	nvoFanSpeedCtrl .value	nvoFanSpeedCtrl .state
0	0 %	0
1	33,0 %	1
2	66,5 %	1
3	100 %	1

UCPTfanSpdStages = 11

Lüfterstufe	nvoFanSpeedCmd .value	nvoFanSpeedCmd .state
AUTO	0 %	-1
0	0 %	0
1	100 %	1

UCPTfanSpdStages = 12

Lüfterstufe	nvoFanSpeedCmd .value	nvoFanSpeedCmd .state
AUTO	0 %	-1
0	0 %	0
1	50 %	1
2	100 %	1

UCPTfanSpdStages = 13

Lüfterstufe	nvoFanSpeedCmd .value	nvoFanSpeedCmd .state
AUTO	0 %	-1
0	0 %	0
1	33,0 %	1
2	66,5 %	1
3	100 %	1

### nvoSpaceHumidity

SNVT Typ: SNVT\_levPercent, Index 95

Function: Output Variable for indication of relative humidity (only devices with humidity).  
Data output is made after a change of the value, depending on *SCPTmaxSendTime* and 1,5s- 4s after Reset.

## 7.3 Configuration Property Space Comfort Controller Object - General:

### SCPT HVAC Type

Index: 169, SNVT\_hvac\_type

Function: Configuration property for identification of a controller type.

Preset value: nciHvacType = HVT\_GENERIC

### SCPT HVAC Mode

Index: 74, SNVT\_hvac\_mode

Function: The configuration property determines the initialisation status of the input variable nviApplicMode and thus also the start configuration of the temperature controller. Preset value: HVAC\_AUTO

### SCPTmaxSendTime

Index: 49, SNVT\_time\_sec

Function: The configuration property defines the interval time for the calculation of new control variables for the temperature control and the sending time of the output variable. By input values = 0, data output is deactivated. (Preset value: 30 s)

### SCPT Offset Temp

Index: 70, SNVT\_temp\_p

Function: Offset for the temperature value. By this parameter a software calibration is possible.

### SCPToffsetRH

Index: 69, SNVT\_lev\_percent

Function: Offset for the humidity value. By this parameter a software calibration is possible.

### SCPTminSendTime

Index: 52, SNVT\_time\_sec

Function: Stipulates the smallest update interval of the output variable nvoSpaceTemp. An update is made after expiration of „SCPTminSendTime“, if the temperature value of the output variable has changed by more than „SCPTminDeltaTemp“. By means of the input values = 0 the function is deactivated.  
(Preset value: 5,0 sec)

### SCPTminDeltaTemp

Index: 64, SNVT\_temp\_p

Function: If the temperature has changed by the adjusted value „SCPTminDeltaTemp“ the new temperature value is transmitted. The function is depending on the adjustment of the property „SCPTminSendTime“. (Range >= 0 °C; preset value: 0,30 °C)

### UCPTstring4

Index: 4, SNVT\_str\_asc

Function: With this property the text for the unit (max. 3 ASCII signs) and the description (max. 4 ASCII signs) of the controller set point are set. These texts are displayed when the set point is faded-in.

### SCPTnumValves

Index: 59, SNVT\_count

Function: The configuration property is designed for the selection of 2 or 4 tube systems. If a 2 tube system is operated (1 valve), the output variables including the control variables for heating and cooling receive the same values.

**SCPTnumValves = 1: ==> 2-tube-system**

Mode heating: nvoHeatPrimary = nvoCoolPrimary = control variable heating

Mode cooling: nvoHeatPrimary = nvoCoolPrimary = control variable cooling

**SCPTnumValves = 2: ==> 4-tube-system (standard value)**

Mode heating: nvoHeatPrimary = control variable heating

Mode cooling: nvoCoolPrimary = control variable cooling

### UCPTpwmCycleTime

Index: 35, SNVT\_time\_min

Function: The configuration property determines the cycle time for a pulse width modulated control of the control variables by nvoHeatPWM and nvoCoolPWM. (Preset value: 15 min)

### SCPTbypassTime

SCPT Index: 34, SNVT\_time\_min

Function: Delay time in minutes. After expiration of SCPTbypassTime the Output Variable nvoOsOccup is set back to OC\_UNOCCUPIED. (Preset value: 0 min). It should be noted that with a value > 0 every trigger of the presence key restarts the bypass time. To use the occupancy toggle function the value should be set to 0.

### UCPTgeneralCP1

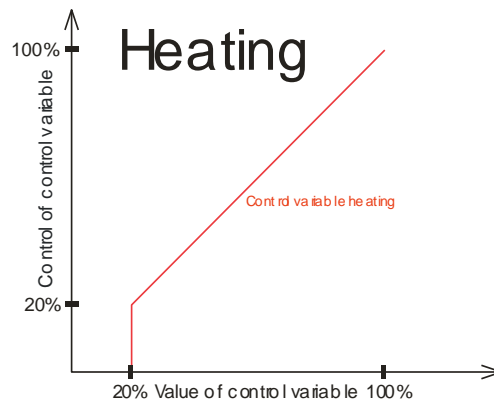
Index: 7, SNVT\_state

Function: The configuration parameter indicates the behaviour of the node.

UCPTgeneralCP1.bit[0] = 1 ==> **Minimal actuating variable** actuating variable > 0

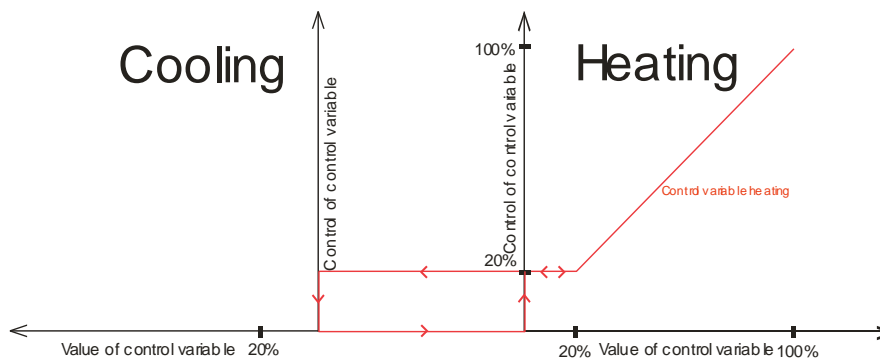
UCPTgeneralCP1.bit[0] = 0\* ==> **Minimal actuating variable** actuating variable = 0

- (1) Mode selection control variable = 1  
Ymin = 20%



The control variable is only sent to the output if the calculated value of the control variable is bigger than the minimal control variable.

- (2) mode selection control variable = 0  
Ymin = 20%



The minimal control variable at the output remains unchanged until the controller changes the operating mode

\* = Preset values

### UCPTrhSpOffsetRg

Index: 17, SNVT\_lev\_percent

Function: If the humidity has changed by the adjusted value „SCPTminDeltaTemp“, the new humidity value is transmitted. (Range  $\geq 0$  %; preset value: 3%)

## 7.4 Configuration Property Space Comfort Controller Object - Set Point:

### SCPTsetPnts

Index: 60, SNVT\_temp\_setpt

Function: Configuration property for default of the set points for heating and cooling depending on the room occupancy. By nviSetpoint the values can be overwritten by nvoOccupEffect = OCCUPIED respectively STANDBY. When having nvoOccupEffect = UNOCCUPIED, nviSetpoint is not considered, however.

Preset values:	.occupied_heat 21,00 °C	.occupied_cool 23,00 °C
	.standby_heat 19,00 °C	.standby_cool 25,00 °C
	.unoccupied_heat 16,00 °C	.unoccupied_cool 28,00 °C

### **UCPTspAntiFreez**

Index: 18, SNVT\_temp\_p

Function: Set point for heating for antifreeze function with window contact opened, i.e. with an active energy-saving function. (Preset value: 10°C)

### **UCPTspEditDelta**

Index: 9, SNVT\_temp\_p

Function: Configuration property for definition of temperature jumps via the operating keys. (Preset value: 0.5K, i.e. the set point is changed by 0,5K with any key actuation).

### **SCPTminRnge, SCPTmaxRnge**

Index: 23, SNVT\_temp\_p, 20, SNVT\_temp\_p

Function: Configuration parameter for the value range of the adjusted set point correction, i.e. the prescribed set point can be changed by the user by means of the value +/- SCPTmaxRnge / SCPTminRnge. (Preset value: 3,0 K)

## **7.5 Configuration Property Space Comfort Controller Object - Fan Coil Unit:**

### **UCPTfanSpdStages**

Index: 13, SNVT\_count

Function: Configuration property for default of fan stages.  
(Preset value: 3 ==> AUTO, OFF, Stage 1, Stage 2, Stage 3)

### **UCPTfanTiming**

Index: 33, UNVT\_fan\_timing

Function: Configuration property for default of minimal switch-on time of fan  
UNVT\_fan\_timing.MinOnTime: minimal switch-on time of a fan stage (initialisation value: 120 sec.)

### **UCPTfanStageLev**

Index: 36, UNVT\_fan\_stg\_lev

Function: Configuration property for default of switch value of fan stages for heating and cooling. (see function diagram controller)

Preset value:	.CoolFirstStage	0,000 %
	.CoolSecondStage	33,000 %
	.CoolThirdStage	66,500 %
	.HeatFirstStage	0,000 %
	.HeatSecondStage	33,000 %
	.HeatThirdStage	66,500 %

### **UCPTfanAccByCtr**

Index: 81, typedef enum {MEM\_NUL;DISABLE;ENABLE}th\_on\_off\_t

Funktion: Selection if the fan of controller has access to the „main“ fan.



## 7.6 Configuration Property Space Comfort Controller Object - PID-Controller Heating:

### UCPTheatXp

Index: 19, SNVT\_temp\_p

Function: Property for adjusting the proportional range. By UCPTheatXp = 0 the controller for heating is deactivated. (Preset value: 4K, range: 0-10 K)

### UCPTheatTn

Index: 20, SNVT\_time\_min

Function: Property for adjusting the reset time of I-proportion (scanning time Ta= SCPTmaxSendTime). By input values = 0, the I-proportion is deactivated. (Preset value: 100 min)

### UCPTheatMinDay

Index: 27, SNVT\_lev\_percent

Function: Control variable limit downwards in the operation mode OCCUPIED and STANDBY.  
(Preset value: 0 %)

### UCPTheatMax

Index: 28, SNVT\_lev\_percent

Function: Control variable limit upwards.(Preset value: 100 %)

## 7.7 Configuration Property Space Comfort Controller Object - PID-Controller Cooling:

### UCPTcoolXp

Index: 22, SNVT\_temp\_p

Function: Property adjusting the proportional range. By UCPTheatXp = 0 the controller for heating is deactivated.  
(Preset value: 4 K, range: 0-10 K)

### UCPTcoolTn

Index: 23, SNVT\_time\_min

Function: Property for adjusting the reset time of the I-proportion (scanning time Ta = SCPTmaxSendTime). By input values = 0, the I-proportion is deactivated. (Preset value: 100 min)

### UCPTcoolMinDay

Index: 30, SNVT\_lev\_percent

Function: Control variable limit downwards in the operation mode OCCUPIED and STANDBY.  
(Preset value: 0 %)

### UCPTcoolMax

Index: 31, SNVT\_lev\_percent

Function: Control variable limit upwards. (Preset value: 100 %)