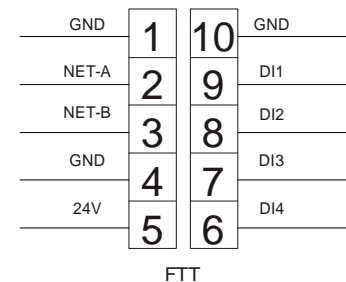


# Software Description

Subject to technical alteration  
Issue 26.05.11

## WRF06LCD 2011



## 1 Overview

The application for the room operating panel WRF06LCD can take over the functions temperature control with PI-controller for heating/cooling, temperature detection, set point adjustment, presence button with overtime function and fan stage adjustment. The defaults of the LonMark® function profiles **8500 „Space Comfort Controller“** are considered. For extended setting options, user defined configuration properties (UCPT) are used. The UCPTs used are defined in the **Thermokon Device Recource Files** from version 2.1 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 as well as by pressing a menu button on the top of the device. If the device was decommissioned, a voltage reset must be made if the device shall be commissioned again by pressing a menu button.

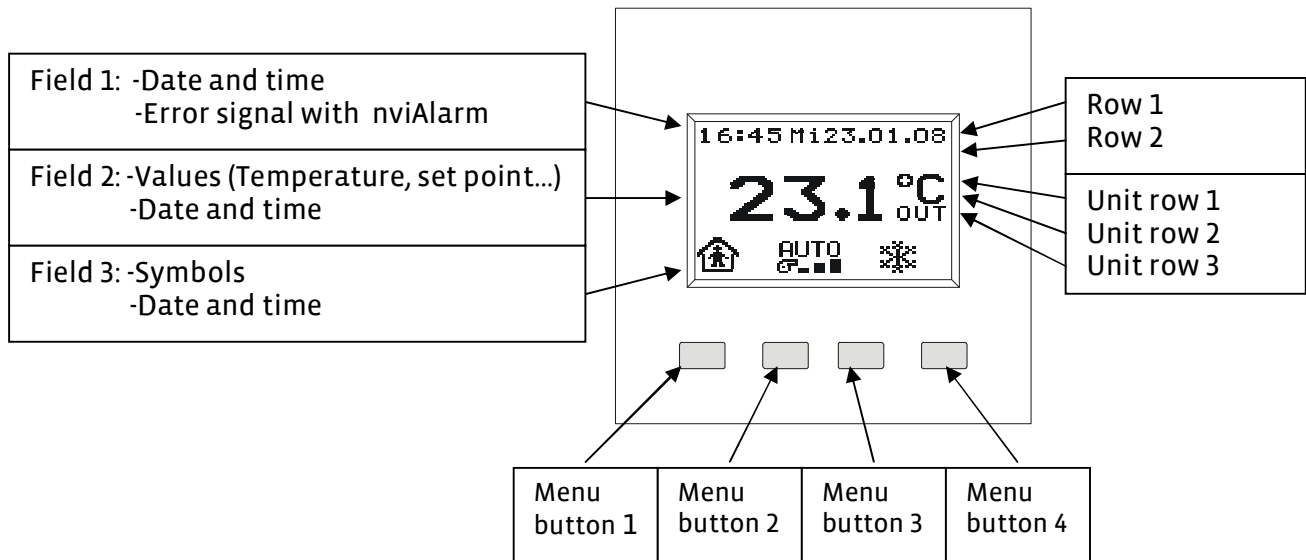
### 1.2. Inputs

The device has 4 dry inputs which are free for parametrizing e.g. for

- Switching lighth
- Dew point, energy hold off
- room occupancy

## 1.2 Upper Part

The display is divided into 3 display areas: field 1 in the upper range, field 2 in the middle and field 3 in the bottom range.



The symbols displayed and their functions are as follows:

 Set point adjustment

 Error

 Heating

 Cooling

 Window "open"

 De point detector "active"

 Occupied (comfort) / Unoccupied (stand-by)

 Occupied (Extension of party time)

 Fan off

 Fan stage 1

 Fan stage 2

 Fan stage 3

 Fan off

 Fan stage 1

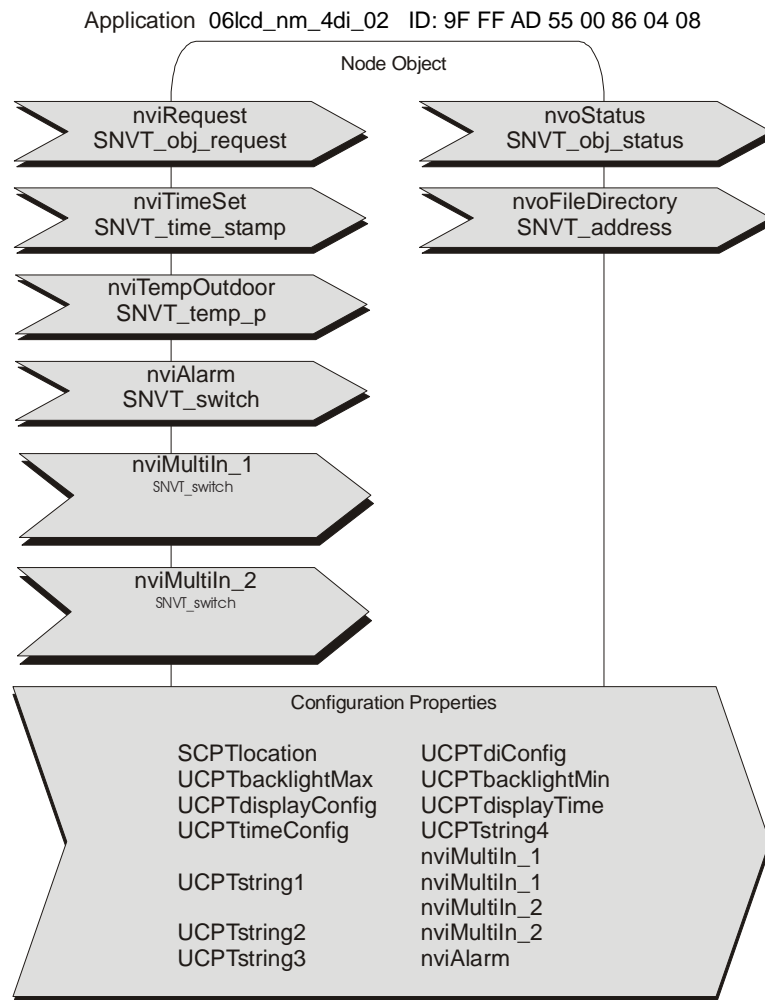
 Fan stage 2

 Fan stage 3

If menu buttons 1 and 4 are pressed at the same time for approx. 5 seconds, the calibration mode of the device is polled and an offset for the temperature of the device can be set.

## 2 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.

### Menu Buttons:

By UCPTdiConfig the menu buttons 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. after a module reset or if no button actuation is made for more than 15 sec., the value UCPTbacklightMin determines the brightness of the LCD. By button actuation the stand-by mode is left and the value UCPTbacklightMax determines the brightness of the background illumination.

### 2.1 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 LCD symbol „failure“:

nviAlarm = 100.0 1 ==> symbol „ failure“ on

nviAlarm = 0.0 0 ==> symbol „ failure“ off

By the configuration property UCPTString3 the error signal is defined. The failure message is indicated in display 1.

**nviTempOutdoor**

SNVT Type: SNVT\_temp\_p, Index 105

Function: By this input variable the outdoor temperature can be prescribed additionally.

**nviMultiIn\_1; nviMultiIn\_2**

SNVT Type: SNVT\_switch

Function: Depending on the configuration the input variable is indicated in the display.

Property: By the property UCPTString1 respectively UCPTString2 the unit is prescribed.

**2.2 Output Variable 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.

**2.3 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.  
 UCPTtimeConfig[1].Byte[0...3] configures time and date in the main field of the display.  
 !! Please note: After a module reset the display for date and time is faded out. The time is only  
 !! faded in after receipt of a telegram via the input variable nviTimeSet. The time should be  
 !! synchronized by a LON system clock once a day.

UCPTtimeConfig.Byte[0] = 0 ==> **Weekday** faded out  
 UCPTtimeConfig.Byte[0] = 1 ==> **Weekday** English  
 UCPTtimeConfig.Byte[0] = 2 ==> **Weekday** German  
 UCPTtimeConfig.Byte[1] = 0 ==> **Date** faded out  
 UCPTtimeConfig.Byte[1] = 1 ==> **Date** English (JJ.MM.TT)  
 UCPTtimeConfig.Byte[1] = 2 ==> **Date** German (TT.MM.JJ)  
 UCPTtimeConfig.Byte[2] = 0 ==> **Time** faded out  
 UCPTtimeConfig.Byte[2] = 1 ==> **Time** 24-hours mode  
 UCPTtimeConfig.Byte[2] = 2 ==> **Time** 12-hours mode  
 Preset values: .Byte[0] = 2, .Byte[1] = 2, .Byte[2] = 1

UCPTtimeConfig[1].Byte[0] = 0 ==> **Date** English (MM.TT)  
 UCPTtimeConfig[1].Byte[0] = 1 ==> **Date** German (TT.MM)  
 UCPTtimeConfig[1].Byte[1] = 0 ==> **Time** 24-hours mode  
 UCPTtimeConfig[1].Byte[1] = 1 ==> **Time** 12-hours mode  
 Present values: .Byte[0] = 1, .Byte[1] = 0

## Node Object

**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)

**UCPTbacklightMin**

UCPT Index: 48, SNVT\_lev\_cont

Function: In the stable state, i.e. after module reset or if no button actuation is made for a period longer than 15 sec. the value UCPTbacklightMin determines the background illumination of the LCD.

**UCPTbacklightMax**

UCPT Index: 47, SNVT\_lev\_cont

Function: If the stable state is left by actuating a button, the value UCPTbacklightMax determines the brightness of the background illumination of the LCD.

**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.bit[0] = 1\* ==> **Room temperature** displayed

UCPTdisplConfig.bit[0] = 0 ==> **Room temperature** not displayed

UCPTdisplConfig.bit[1] = 1 ==> **Outdoor temperature** displayed

UCPTdisplConfig.bit[1] = 0\* ==> **Outdoor temperature** not displayed

UCPTdisplConfig.bit[2] = 1 ==> **Set temperature effective** displayed

UCPTdisplConfig.bit[2] = 0\* ==> **Set temperature effective** not displayed

UCPTdisplConfig.bit[3] = 1 ==> **Set point offset** displayed

UCPTdisplConfig.bit[3] = 0\* ==> **Set point offset** not displayed

If bit2 = 1 and bit3 = 1, then the effective set point temperature is displayed

UCPTdisplConfig.bit[4] = 1 ==> **nviMultiIn\_1** displayed

UCPTdisplConfig.bit[4] = 0\* ==> **nviMultiIn\_1** not displayed

UCPTdisplConfig.bit[5] = 1 ==> **nviMultiIn\_2** displayed

UCPTdisplConfig.bit[5] = 0\* ==> **nviMultiIn\_2** not displayed

UCPTdisplConfig.bit[6] = 1 ==> **Time** displayed

UCPTdisplConfig.bit[6] = 0\* ==> **Time** not displayed

UCPTdisplConfig.bit[7] = 1 ==> **Date** displayed

UCPTdisplConfig.bit[7] = 0\* ==> **Date** not displayed

UCPTdisplConfig.bit[8] = 1\* ==> **°C** is the unit of measurement for the temperature display

UCPTdisplConfig.bit[8] = 0 ==> **°F** is the unit of measurement for the temperature display

UCPTdisplConfig.bit[9] = 1\* ==> **Decimal point for temperature** displayed

UCPTdisplConfig.bit[9] = 0 ==> **Decimal point for temperature** not displayed

UCPTdisplConfig.bit[10] = 1\* ==> **Decimal point for nviMultiIn\_1** displayed

UCPTdisplConfig.bit[10] = 0 ==> **Decimal point for nviMultiIn\_1** not displayed

UCPTdisplConfig.bit[11] = 1\* ==> **Decimal point for nviMultiIn\_2** displayed

UCPTdisplConfig.bit[11] = 0 ==> **Decimal point for nviMultiIn\_2** not displayed

UCPTdisplConfig.bit[12] = 1\* ==> **Symbols room occupancy** displayed (status of nvoEffectOccup)

UCPTdisplConfig.bit[12] = 0 ==> **Symbols room occupancy** not displayed

UCPTdisplConfig.bit[13] = 1 ==> **Symbols fan** displayed

UCPTdisplConfig.bit[13] = 0\* ==> **Symbols fan** not displayed

UCPTdisplConfig.bit[14] = 1 ==> **Display set point offset** with set point adjustment

UCPTdisplConfig.bit[14] = 0\* ==> **Set temperature effective** with set point adjustment

UCPTdisplConfig.bit[15] = 1 ==> **Russian letters** will be used in textfields

UCPTdisplConfig.bit[15] = 0\* ==> **Latin letters** will be used in textfields

\* = present values

## Node Object

**UCPTdiConfig[0]...[1]**

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

Function: The following configuration properties determine the menu button functions and the digital input function.

UCPTdiConfig[0].Byte[0] configures **Menu button 1**UCPTdiConfig[0].Byte[1] configures **Menu button 2**UCPTdiConfig[0].Byte[2] configures **Menu button 3**UCPTdiConfig[0].Byte[3] configures **Menu button 4**UCPTdiConfig[1].Byte[0] configures **digital input 1**UCPTdiConfig[1].Byte[1] configures **digital input 2**UCPTdiConfig[1].Byte[2] configures **digital input 3**UCPTdiConfig[1].Byte[3] configures **digital input 4**

Preset values: [0].Byte[0] = 0x01, [0].Byte[1] = 0x02, [0].Byte[2] = 0x21, [0].Byte[3] = 0x00

[1].Byte[0] = 0x40, [1].Byte[1] = 0x40, [1].Byte[2] = 0x40, [1].Byte[3] = 0x40

UCPTdiConfig, Configuration of Menu Buttons	
Allocation Button – Function Block	
Byte[0...3]	Description
	<b>No Function</b>
0x00	Unoccupied
	<b>Set point adjustment</b>
0x01	Set temperature Plus
0x02	Set temperature Minus
	<b>Fan Stage</b>
0x10	Plus by AUTO
0x11	Minus by AUTO
0x12	Plus without AUTO
0x13	Minus without AUTO
0x14	Plus with AUTO toggle
0x15	Minus with AUTO toggle
0x16	Plus without AUTO toggle
0x17	Minus without AUTO toggle
0x18	Only AUTO
	<b>Room Occupancy</b>
0x20	With overtime function
0x21	With toggle function
0x22	Only Occupied
0x23	Only UnOccupied
	<b>Value Display</b>
0x30	Indoor temperature - nviSpaceTemp respect. nvoSpaceTemp
0x31	Outdoor temperature - nviTempOutdoor
0x32	Effective set point - nvoSetptEffect
0x33	Offset set point - nvoSetptOffset
0x34	Individual value - nviMultiIn_1

## Node Object

0x35	Individual value - nviMultiIn_2
0x36	Time
0x37	Date
	<b>Switch Function</b>
0x40	active / inactive – make contact
0x41	active / inactive – break contact
0x42	Light toggle – make contact
0x43	Light toggle – break contact
0x44	Command automatic (= 0.0 –1) - make contact
0x45	Command automatic (= 0.0 –1) - break contact

**SCPTnvType**

SCPT Index: 254, SNVT\_nv\_type

Function: One SCPTnvType is available for nviMultiIn\_1 and nviMultiIn\_2 each. This configuration property specifies the type of the network variable nviMultiIn\_1 respectively nviMultiIn\_2. If SCPTnvType is not automatically adapted to the new variable type of nviMultiIn\_1 / nviMultiIn\_2 by the installation tool, the following settings must be input manually. Further values can be found in the Echelon SNVT-Master-List.

nviMultiIn = SNVT\_switch

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

nvoMultiOut = SNVT\_ppm

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

nvoMultiOut = SNVT\_lev\_percent

==&gt; SCPTnvType = PID 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

==&gt; SCPTnvType = PID 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\_press

==&gt; SCPTnvType = PID 0:0:0:0:0:0:0, Scope 0, Index 30, NVT\_CAT\_SIGNED\_LONG, 1 bytes, A=1, B=-1, C=0

**UCPTstring1, UCPTstring2, UCPTstring3**

By the configuration properties alarm texts and units can be put in. The texts are displayed in two different type sizes. By means of the configuration properties alarm texts and units can be entered. The texts are displayed in two different sizes:

Type size 1 -&gt; smallest size

Type size 2 -&gt; medium size

Depending on the type size more or less letters are fitting into the display.

**UCPTstring1[0],[1],[2]**

SCPT Index: 1, SNVT\_str\_asc

Function: Unit for nviMultiIn\_1. If UCPTstring1[1] is empty, UCPTstring1[0] is automatically displayed in type size 2. Otherwise, the unit of type size 1 is displayed. UCPTstring1[2] is always displayed in type size 1. In type size 1 up to three signs and in type size 2 one sign as well as the degree symbol can be displayed.

**UCPTstring2[0],[1],[2]**

SCPT Index: 2, SNVT\_str\_asc

Function: Unit for nviMultiIn\_2. If UCPTstring2[1] is empty, UCPTstring2[0] is automatically displayed in type size 2. Otherwise, the unit is shown in type size 1. UCPTstring2[2] is always displayed in type size 1. In type size 1 up to three signs and in type size 2 one sign as well as the degree symbol can be displayed.

**UCPTstring3[0],[1]**

SCPT Index: 3, SNVT\_str\_asc

Function: Alarm message, which is polled by nviAlarm = 100.0 1.. If UCPTstring3[1] is empty, UCPTstring3[0] is automatically displayed in type size 2. Otherwise, the alarm message is displayed in type size 1. In type size 1 up to 12 signs and in type size 2 up to 7 signs can be displayed.

Node Object

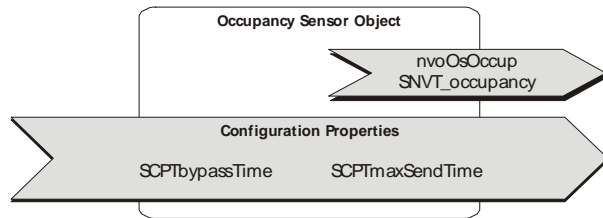
UCPTstring4[0],[1]

SCPT Index: 4, SNVT\_str\_asc

Function: Textfield, which is shown after a reset in row 1 and row 2. If UCPTstring4[1] is empty, UCPTstring4[0] is automatically displayed in type size 2. Otherwise, the text is displayed in type size 1. In type size 1 up to 14 signes and in type size 2 up to 9 signs can be displayed.



### 3 Occupancy Sensor Object



#### Overtime Function with Presence Button:

If a menu button respectively a digital input are configured with presence, the output nvoOsOccup receives the value OCCUPIED upon confirmation. After expiration of the SCPTbypassTime it is reset to the value UNOCCUPIED. For realization of the overtime function and for indication in the display, nvoOsOccup must be binded to the input variable nviOccSensor.

The Occupancy Sensor Object can be switched by a conventional occupancy sensor. The occupancy sensor shall be connected to a digital input. Via UCPTdiConfig[1] (in NodeObject) the function for a motion detection is adjusted.

#### 3.1 Output Variables Occupancy Sensor Object:

##### nvoOsOccup

SNVT Type: SNVT\_occupancy, Index 109

Function: Output variable for presence detection in rooms. The output values are depending on the function settings. By UCPTdiConfig.Byte[0...3] = the menu buttons are allocated to the Occupancy-Sensor-Object. UCPTdiConfig.Byte[0...3] = 20<sub>hex</sub> - 23<sub>hex</sub> allocates a function to these buttons. Several buttons can be allocated to an Occupancy-Sensor-Object. For local presence detection nvoOsOccup can also be binded to the input variable nviOccSensor.

##### UCPTdiConfig.Byte[0...3] = 20<sub>hex</sub>, Presence button with overtime function

By button actuation the output variable nvoOsOccup receives the value OC\_OCCUPIED and the transit time is started. After expiration of the delay time SCPTbypassTime the output variable is reset to the value UNOCCUPIED. Each button actuation restarts the tracking time.

##### UCPTdiConfig.Byte[0...3] = 21<sub>hex</sub>, Presence button with toggle function

By button actuation the output variable nvoOsOccup is toggled between the values OC\_OCCUPIED and OC\_UNOCCUPIED.

##### UCPTdiConfig.Byte[0...3] = 22<sub>hex</sub>, Presence button OCCUPIED

By button actuation the output variable nvoOsOccup receives the value OC\_OCCUPIED.

##### UCPTdiConfig.Byte[0...3] = 23<sub>hex</sub>, Presence button UNOCCUPIED

By button actuation the output variable nvoOsOccup receives the value OC\_UNOCCUPIED.

#### 3.2 Configuration Property Occupancy Sensor Object:

##### SCPTbypassTime

SCPT Index: 34, SNVT\_time\_min

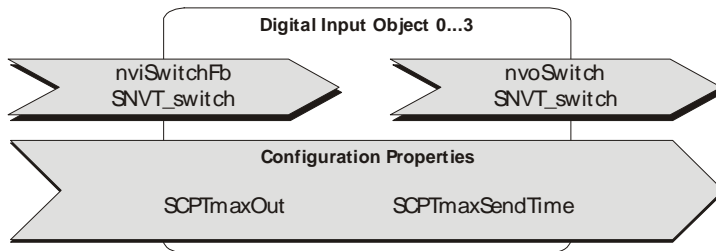
Function: Delay time in minutes. After expiration of SCPTbypassTime the output variable nvoOsOccup is set back to OC\_UNOCCUPIED. (Preset value: 90 min)

##### SCPTmaxSendTime

Index: 49, SNVT\_time\_sec

Function: Heartbeat function. This configuration property stipulates the interval time after which the output variables are sent. By means of input values =0, the heartbeat function is deactivated. (Preset value: 120 s)

## 4 Digital Inputs



The status of the four potential-free digital inputs is detected and output by the output variables type `SNVT_switch` and `SNVT_setting`, depending on the configuration (`UCPTdiConfig`), whereas with `SNVT_switch` an absolute light value for manual override is sent

The digital inputs can be set by the following functions:

- Switch function:toggle, status
  - For light switch, window contact, dew point detector, air-flow monitoring
- Occupancy sensor

### 4.1 Input Variable Digital Input Object:

#### `nviSwitchFb_x`

SNVT Type: `SNVT_switch`, Index 95

Function: Input variable for current status of the light groups controlled by `nvoSwSwitch`.

### 4.2 Output Variable Digital Input Object:

#### `nvoSwitch_x`

SNVT Typ: `SNVT_switch`, Index 95

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

#### **Standard I/O:**

Potential-free contact active ==> `nvoSwSwitch.state = 1`

`nvoSwSwitch.value = SCPTmaxOut`

Potential-free contact inactive ==> `nvoSwSwitch = 0.0 0`

#### **Toggle:**

Toggle inactive ==> active ==> Each button actuation results in toggling of the variables between ON and OFF.

Lighting ON `nvoSwSwitch.state = 1`

`nvoSwSwitch.value = SCPTmaxOut`

Lighting OFF `nvoSwSwitch1/2 = 0.0 0`

#### **Befehl Automatik:**

Wechsel *nicht aktiv* ==> *aktiv* ==> Bei Tastbetätigung wird die Ausgangsvariable `nvoSwitch` mit dem Wert 0.0 -1 gesendet. Die Ausgangsvariablen werden nach Wechsel des Ausgabewertes, nach Ablauf der Heartbeat-Zeit (`SCPTmaxSendTime`) und 1,5s- 4s nach Modul-Reset ausgegeben.

#### **Command Automatic:**

Toggle inactive ==> active ==> Upon button actuation the output variable `nvoSwSwitch` is sent with the value 0.0 -1. The output variables are output after change of the output values, after expiration of the heartbeat time (`SCPTmaxSendTime`) and 1,5s- 4s after module reset.

### 4.3 Configuration Parameter Digital Input Object:

#### `SCPTmaxOut`

SCPT Index: 93, `SNVT_lev_cont`

Function: Configuration parameter to determine the maximal output value of the variable `nvoSwitch.value`. (Present Value: 100 %)

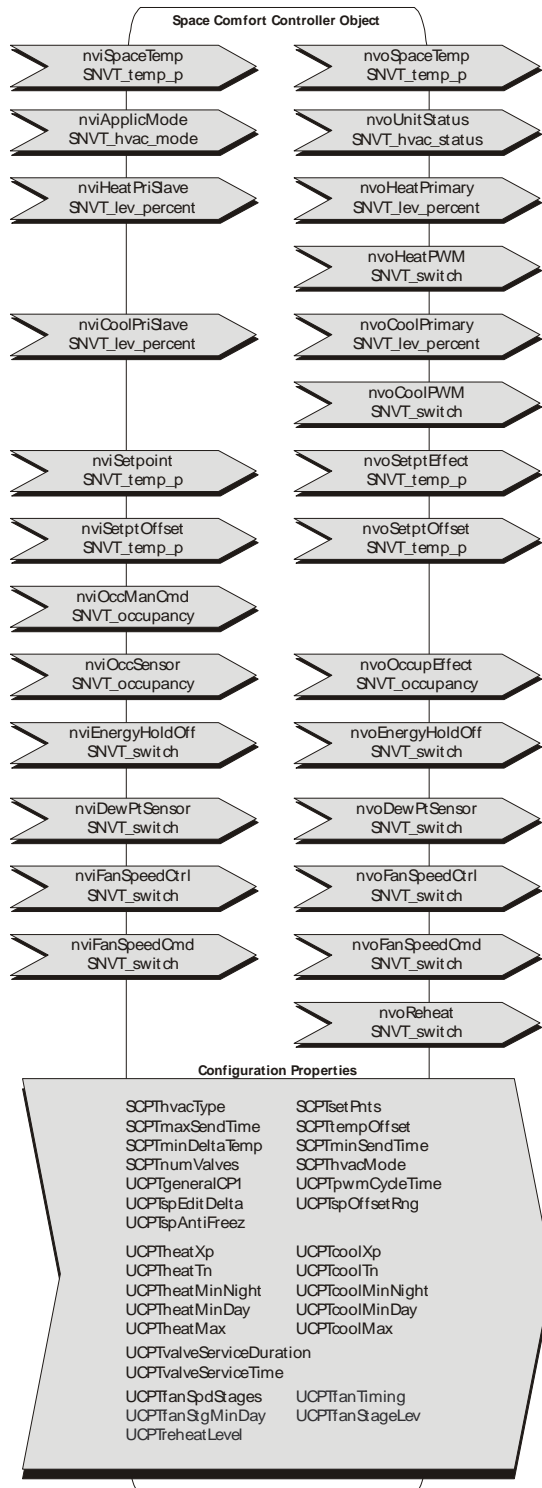
#### `SCPTmaxSendTime`

SCPT Index: 49, `SNVT_time_sec`

Function: Heartbeat interval. After expiration of the time `SCPTmaxSendTime` the digital input is polled and the output variables are up-dated. By input values = 0 the heartbeat function is deactivated. (Preset value: 0)

## 5 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 buttons. Output of the offset value is made by nvoSetptOffset.

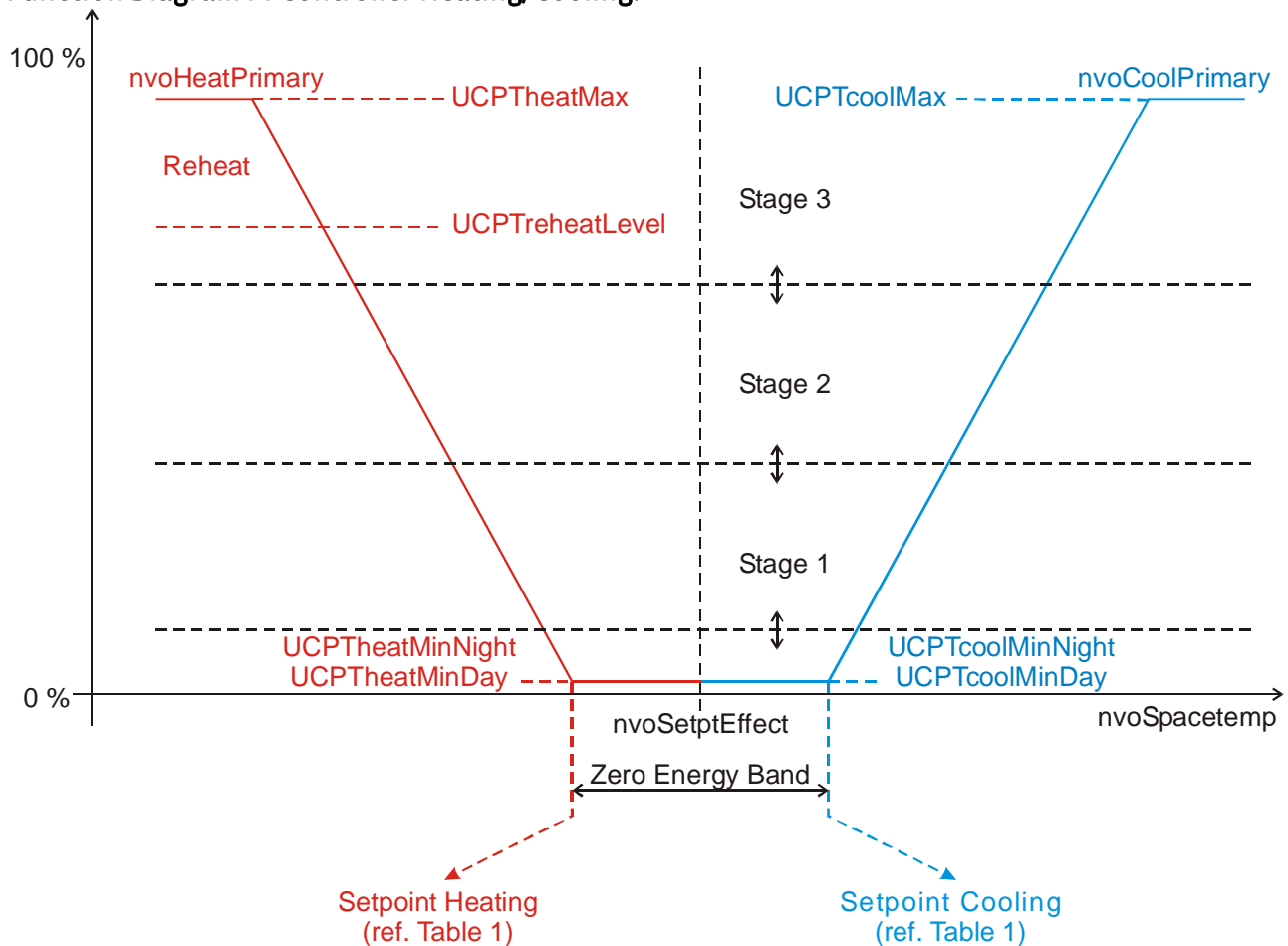
The effective set point (basic set point) nvoSetptEffect is calculated in reliance on the input variables to the room occupancy (nviOccManCmd and nviOccSensor) of the set point defaults SCPTsetPnts respectively nviSetpoint and the set point offset.

**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 parameterizable 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.

## Space Comfort Controller

## Function Diagram PI-Controller Heating/Cooling:



## 5.1 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 %)` ==> internal controller cooling ON (initialisation value after reset).  
`nviCoolPriSlave = 0...100 %` ==> internal controller cooling OFF  
 ==> `nviCoolPriSlave` determines the output variables.  
 !! The external override has top priority, thus also a concurrent control of the heating and  
 !! cooling valve is possible.

**nviHeatPriSlave**

SNVT Type: SNVT\_lev\_percent, Index 81

Function: Control variable for network variable nvoHeatPrimary and nvoHeatPWM.  
 nviHeatPriSlave = 0x7FFF (163,835 %) ==> internal controller heating ON (initialisation value after reset)  
 nviHeatPriSlave = 0 ... 100 % ==> internal controller heating OFF  
 ==> nviHeatPriSlave determines the output quantities  
 !! The external override has top priority, thus also a concurrent control of the heating and  
 !! cooling valve is possible.

**nviOccManCmd und nviOccSensor**

SNVT Type: 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 via building control technology: OC\_OCCUPIED, OC\_BYPASS, OC\_STANDBY, OC\_UNOCCUPIED  
 nviOccSensor: presence status message in rooms: OC\_OCCUPIED, OC\_UNOCCUPIED  
 Via the configuration property *UCPTgeneralCPI* the behaviour of the device can be determined if nviOccManCmd = OC\_UNOCCUPIED is switched.

nviOccManCmd	nviOccSensor		room occupancy nvoOccupEffect	Set Point Heat nvoSetptEffect (nvoUnitStatus.mode = HVAC_HEAT)	Set Point Cool nvoSetptEffect (nvoUnitStatus.mode = HVAC_COOL)
OC_NUL	OC_NUL	>>>	OCCUPIED	SCPTsetPnts.occupied_heat + nviSetptOffset	SCPTsetPnts.occupied_cool + nviSetptOffset
OC_OCCUPIED	****	>>>		or nviSetptOffset + nviSetpoint - ( SCPTsetPnts.occupied_cool - SCPTsetPnts.occupied_heat ) / 2	or nviSetptOffset + nviSetpoint + ( SCPTsetPnts.occupied_cool - SCPTsetPnts.occupied_heat ) / 2
****	OC_OCCUPIED	>>>			
OC_STANDBY	OC_NUL OC_UNOCCUPIED	>>>	STANDBY	SCPTsetPnts.standby_heat + nviSetptOffset or nviSetptOffset + nviSetpoint - ( SCPTsetPnts.standby_cool - SCPTsetPnts.standby_heat ) / 2	SCPTsetPnts.standby_cool + nviSetptOffset or nviSetptOffset + nviSetpoint + ( SCPTsetPnts.standby_cool - SCPTsetPnts.standby_heat ) / 2
OC_UNOCCUPIED	OC_NUL OC_UNOCCUPIED	>>>	UNOCCUPIED	SCPTsetPnts.unoccupied_heat	SCPTsetPnts.unoccupied_cool

**Table 1: Control porperty depending on the room occupancy****nviFanSpeedCmd**

SNVT Type: SNVT\_switch, Index 95

Function: Input variable to set a new fan speed value for display and nvoFanSpeed.

## Space Comfort Controller

**nviFanSpeedCtrl**

SNVT Type: SNVT\_switch, Index 95

Function: The input variable nviFanSpeedCtrl enables the display of the current fan speed if the externally connected controller selects the fan speed autonomously by nvoFanSpeedCmd 0,0 -1. Therefore, the output variable of the controller (e.g. nvoFanSpeedCmd) must be connected with the input variable nviFanSpeedCtrl.

**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 (seeUCPTspAntiFreez). 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.

## 5.2 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

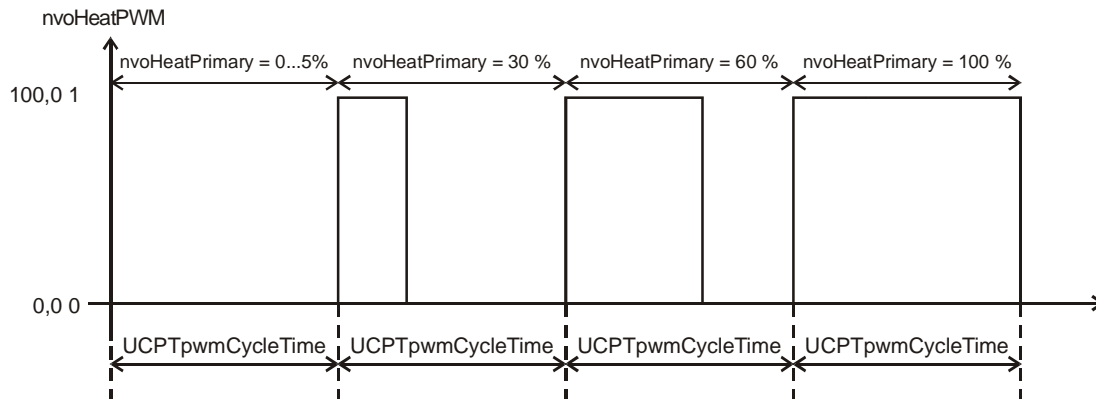
Function: Output variable including the control variable of the PID-controller for heating to control a continuous actuator. Data output is made depending on SCPTmaxSendTime and 1,5s- 4s after reset.

## Space Comfort Controller

**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.



## Space Comfort Controller

**nvoDewPtSensor**

SNVT Type: SNVT\_switch, Index 95

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

nvoDewPtSensor = 0.0 0 ==&gt; dew point detector inactive

nvoDewPtSensor = 100.0 1 ==&gt; 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

UCPTfanSpdStages = 3

UCPTfanSpdStages = 2

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

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

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

!! The acutation of the buttons is only evaluated with nvoEffectOccup = OC\_OCCUPIED. Otherwise, button !!actuactions are ignored.

UCPTdiConfig.Byte[0...3] = 10hex - 18hex (in Node Object) allocates a function for fan speed adjustment to the menu buttons.

**UCPTdiConfig.Byte[0...3] = 10hex, increase fan stage by AUTO**

By button actuation the output variable *nvoFanSpeed* is increased.

**UCPTdiConfig.Byte[0...3] = 11hex, reduce fan stage by AUTO**

By button actuation the output variable *nvoFanSpeed* is reduced.

**UCPTdiConfig.Byte[0...3] = 12hex, increase fan stage without AUTO**

By button actuation the output vairable *nvoFanSpeed* is increased.

**UCPTdiConfig.Byte[0...3] = 13hex, reduce fan stage without AUTO**

By button actuation the output variable *nvoFanSpeed* is reduced.

**UCPTdiConfig.Byte[0...3] = 14hex, increase fan stage with AUTO toggling**

By button actuation the output variable *nvoFanSpeed* is increased. A button actuation at max. fan stage resets the fan stage to AUTO.

**UCPTdiConfig.Byte[0...3] = 15hex, reduce fan stage by AUTO toggling**

By button actuation the output variable *nvoFanSpeed* is reduced. A button actuation with AUTO resets to the maximal stage.

**UCPTdiConfig.Byte[0...3] = 16hex, increase fan stage without AUTO toggling**

By button actuation the outupt variable *nvoFanSpeed* is increased. A button actuation with max. fan stage resets to „0“. The value AUTO is skipped.

**UCPTdiConfig.Byte[0...3] = 17hex, reduce fan stage without AUTO toggling**

By button actuation the output variable *nvoFanSpeed* is reduced. A button actuation with „0“ resets to the maximal stage. The value AUTO is skipped.

**UCPTdiConfig.Byte[0...3] = 18hex, fan stage only AUTO**

By button actuation the output variable *nvoFanSpeed* is toggled to the value 0.0 -1 .



## Space Comfort Controller

**nvoFanSpeedCmd**

SNVT Type: SNVT\_switch, Index 95

Function: Output variable for adjusted fan speed and for concatenation of devices. The number of fan stages is adjustable by means of the configuration property UCPTfanSpdStages. The indication in the display shows the current fan stage adjusted and whether the controller is working in the automatic modue.

 UCPTfanSpdStages = 1  
 = 3

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

UCPTfanSpdStages = 2

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

UCPTfanSpdStages

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

!! Actuation of the buttons is **is only evaluated with nvoEffectOccup = OC\_OCCUPIED**.

!! Otherwise the button actuation is ignored.

**nvoReheat**

SNVT Type: SNVT\_switch, Index 95

Function: Output variable for display of reheating active/  
 nvoReheat = 0.0 0 ==> reheating inactivated  
 nvoReheat = 100.0 1 ==> reheating activated  
 Data output is made after value change depending on *SCPTmaxSendTime* and 1,5s- 4s after reset.

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

**SCPTHvacType**

Index: 169, SNVT\_hvac\_type

Function: Configuration property for identification of a controller type.

Preset value: nciHvacType = HVT\_GENERIC

**SCPTHvacMode**

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)

**SCPTtempOffset**

Index: 272, SNVT\_temp\_diff\_p

Function: Offset for the temperature 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)

## Space Comfort Controller

**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)

**UCPTvalveServiceTime – Maintenance Interval**

Index: 68, SNVT\_time\_hour

Function: The configuration property determines the maintenance interval for the valve. After expiration of the maintenance interval the valve is fully opened and closed again thus protecting the valve and preventing lime deposit.

**UCPTvalveServiceDuration – Period of Maintenance Interval**

Index: 69, SNVT\_time\_min

Function: The configuration property determines the period of time i.e. how long the valve should be opened for the protection of deposit. The time in question should at least guarantee a full opening of the valve.

**UCPTgeneralCP1**

Index: 7, SNVT\_state

Function: The configuration property determines the behaviour of the node nviOccManCmd = OC\_UNOCCUPIED is switched.

UCPTgeneralCP1.bit[0] = 1 ==> reset **nviOccSensor** to „OC\_NUL“UCPTgeneralCP1.bit[0] = 0\* ==>no reset of **nviOccSensor**UCPTgeneralCP1.bit[1] = 1 ==> reset **nvoOsOccup** to „OC\_NUL“UCPTgeneralCP1.bit[1] = 0\* ==>no reset of **nvoOsOccup**UCPTgeneralCP1.bit[2] = 1 ==>reset **Set point offset**UCPTgeneralCP1.bit[2] = 0\* ==> no reset of **Set point offset**UCPTgeneralCP1.bit[3] = 1 ==>reset **fan stage** to „AUTO“UCPTgeneralCP1.bit[3] = 0\* ==>no reset of **fan stage**UCPTgeneralCP1.bit[4] = 1 ==>set **fan stage** to „Off“UCPTgeneralCP1.bit[4] = 0\* ==> no reset of **fan stage**UCPTgeneralCP1.bit[5] = 1 ==>reset **nvoSwitch[0]** to „Off“UCPTgeneralCP1.bit[5] = 0\* ==>no reset of **nvoSwitch[0]**UCPTgeneralCP1.bit[6] = 1 ==> reset **nvoSwitch[1]** to „Off“UCPTgeneralCP1.bit[6] = 0\* ==>no reset of **nvoSwitch[1]**UCPTgeneralCP1.bit[7] = 1 ==>set **nvoSwitch[2]** to „Off“UCPTgeneralCP1.bit[7] = 0\* ==>no reset of **nvoSwitch[2]**UCPTgeneralCP1.bit[8] = 1 ==> reset **nvoSwitch[3]** to „Off“UCPTgeneralCP1.bit[8] = 0\* ==>no reset of **nvoSwitch[3]**

\* = present values

## 5.4 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 buttons. (Preset value: 0.5K, i.e. the set point is changed by 0,5K with any button actuation).

### UCPTspOffsetRng

Index: 12, SNVT\_temp\_p  
 Function: Configuration property 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 +/- UCPTspOffsetRng. (Preset value: 3,0 K)

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

### UCPTreheatLevel

Index: 32, SNVT\_lev\_percent  
 Function: Limiting value for switching-on/-off the electrical reheater. If the control variable nvoHeatPrimary exceeds the value UCPTreheatLevel, the reheater is switched-on (see function diagram controller). (Preset value: 80%)

### 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-in, advance and tracking time of fan  
 UNVT\_fan\_timing.RunUpTime: advance time of fan (initialisation value: 0 sec)  
 UNVT\_fan\_timing.OverrunTime: tracking time of fan (initialisation value: 0 sec)  
 UNVT\_fan\_timing.MinOnTime: minimal switch-on time of a fan stage (initialisation value: 120 sec.)

### UCPTfanStgMinDay

Index: 37, SNVT\_count  
 Function: Configuration property for default of minimal fan stage for the operating modes STANDBY and OCCUPIED. (Preset value Wert: 0, i.e. fan OFF)

### 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 %

## 5.6 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)

### UCPTheatMinNight

Index: 26, SNVT\_lev\_percent

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

### 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 %)

## 5.7 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)

### UCPTcoolMinNight

Index: 29, SNVT\_lev\_percent

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

### 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 %)