

## Softwareapplikation wrf08\_06 (Beleuchtung, Jalousie, Klimasteuerung) Für Fühler Modell WRF08

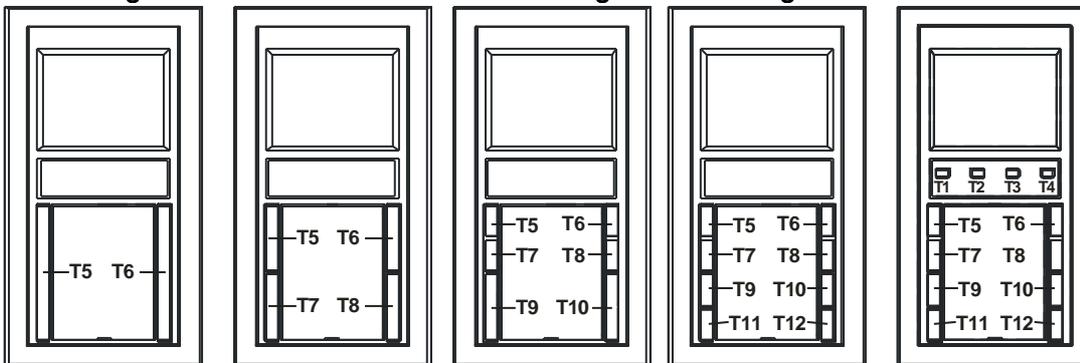
Stand: 10.10.2013

### 1 Übersicht

Das Multifunktions-Raumbediengerät WRF08 dient zur Temperaturerfassung und integrierten Bedienung von HLK, Beleuchtung und Jalousie in der Einzelraumregelung. Die Bedienfunktionen lassen sich flexibel je nach Raumanforderungen verwenden. Hierzu stehen verschiedene Typen mit unterschiedlicher Anzahl an Funktionstasten zur Verfügung.

Die Vorgaben der LonMark® Funktionsprofile **8090** „Space Comfort Control Command Module“, **3200** „Switch“, **1060** „Occupancy Sensor“, und **3250** „Scene Panel“ wurden berücksichtigt. Für erweiterte Einstellmöglichkeiten werden benutzerdefinierte Konfigurationsparameter (UCPT) genutzt. Die hier verwendeten UCPTs sind in den **Thermokon Device Recource Files** ab Version 1.6 oder höher definiert.

#### Darstellung der Gerätevarianten und Beschreibung der LCD-Anzeige:



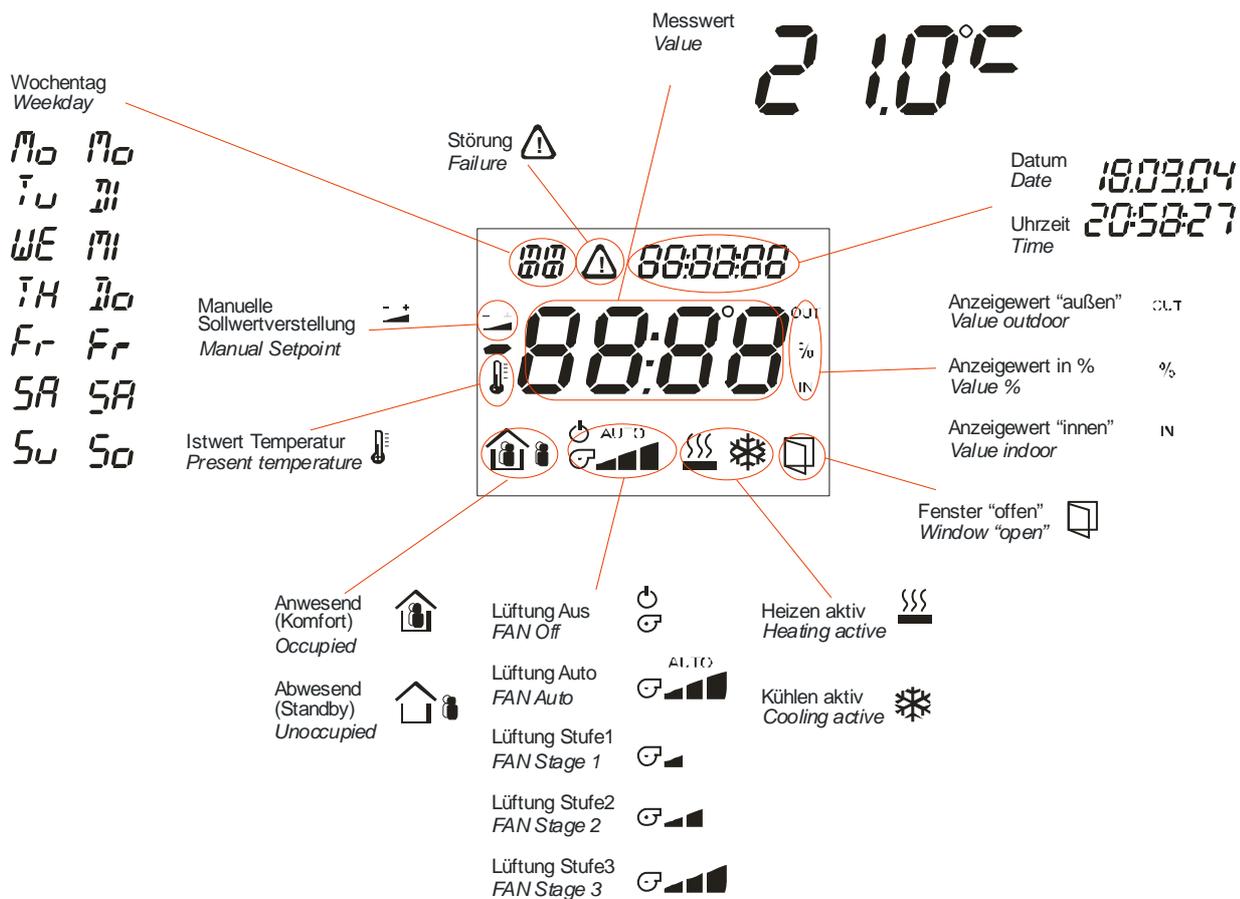
WRF08 2T Standard

WRF08 4T Standard

WRF08 6T Standard

WRF08 8T Standard

WRF08 12T Standard



### Ansteuerung der Segmente

Zeit, Datum, Wochentag: nviTimeSet, UCPTtimeConfig

Messwerte: nvoSpaceTemp, nvoSetptEffect, nvoSetptOffset, nviTempOutdoor, nviRH\_Indoor, nviRH\_Outdoor, UCPTdisplConfig

Raumebelegung: nvoEffectOccup, UCPTdisplConfig

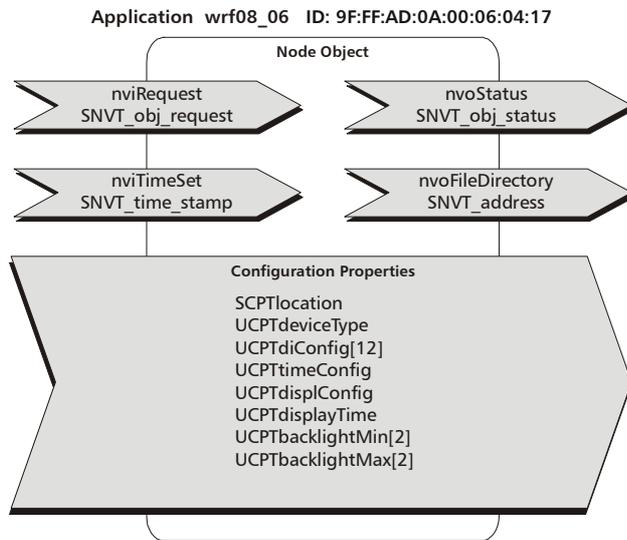
Lüftersymbole: nviFanSpeed, nvoFanSpeed, UCPTdisplConfig

Störung: nviAlarm, nviCmd\_all

Heizen, Kühlen: nviUnitStatus, nviCmd\_all

Fenster: nviEnergyHoldOff, nviCmd\_all

## Node Object



Das Node Objekt überwacht und steuert die Funktionen der einzelnen Objekte im Gerät. Unterstützt wird die von LonMark® geforderte Grundfunktionalität, wobei allgemeine Netzwerkvariablen und Konfigurationsparameter zur Steuerung und Parametrierung des Gerätes eingefügt wurden.

**Uhrzeit / Datum:** Nach Modulreset ist die Anzeige für Datum und Uhrzeit ausgeblendet. Erst nach einer empfangenen Botschaft über die Eingangsvariable *nviTimeSet* wird die Uhrzeit eingeblendet. Die Uhrzeit sollte einmal pro Tag über eine LON-Systemuhr synchronisiert werden. Die Darstellung der Uhrzeit kann mit *UCPTtimeConfig* dem individuellen Kundenwunsch (z.B. Wochentag, Ländereinstellung, ...) angepasst werden. Siehe auch die Parameterbeschreibung auf Seite 4.

**Messwertanzeigen:** Neben der Raumtemperatur können z.B. auch Sollwert, Außentemperatur oder auch rel. Luftfeuchte angezeigt werden. Diese Konfiguration erfolgt über *UCPTdisplConfig*, wobei die Anzeige zwischen den unterschiedlichen Anzeigewerten im Zeitintervall *UCPTdisplayTime* wechselt.

**Hintergrundbeleuchtung:** Die Hintergrundbeleuchtungen von LCD und Beschriftungsfeld können separat angesteuert werden. Im Ruhezustand, d.h. nach Modulreset oder wenn für länger als 15 s kein Tastendruck erfolgt, bestimmt der Wert *UCPTbacklightMin[0]* die Helligkeit des LCD und der Wert *UCPTbacklightMin[1]* die Helligkeit des Beschriftungsfeldes.

Durch Betätigung einer Taste wird der Ruhezustand verlassen und die Werte *UCPTbacklightMax* bestimmen die Helligkeit von LCD und Beschriftungsfeld.

**Tasterfunktionalität und Gerätevarianten:** Während der Inbetriebnahme muss über den Parameter *UCPTdeviceType* der entsprechend verwendete Gerätetyp (2T, 4T, 6T, 8T, 12T) eingestellt werden. Den Tastern können unterschiedliche Funktionen und Objekte wie z.B. Sollwertverstellung oder Beleuchtungssteuerung zugewiesen werden. Mit *UCPTdiConfig[0]* bis *UCPTdiConfig[11]* erfolgt die Konfiguration der Taster T1 bis T12.

## Eingangsvariablen Node Object:

### nviRequest

SNVT Typ: SNVT\_obj\_request, Index 92  
Funktion: Eingangsvariable mit den Funktionen RQ\_NORMAL, RQ\_UPDATE\_STATUS und RQ\_REPORT\_MASK.

### nviTimeSet

SNVT Typ: SNVT\_time\_stamp, Index 84  
Funktion: Eingangsvariable zur Synchronisation der angezeigten Uhrzeit mit einer LON-Systemuhr. Nach einem Modulreset ist die Uhrzeit so lange ausgeblendet bis ein erster gültiger Wert empfangen wurde.

### **Ausgangsvariablen Node Object:**

#### **nvoStatus**

SNVT Typ: SNVT\_obj\_status, Index 93

Funktion: Ausgangsvariable mit den geforderten Status Bits „invalid\_id“ und „invalid\_request“.

#### **nvoFileDirectory**

SNVT Typ: SNVT\_address, Index 114

Funktion: Die Ausgangsvariable stellt dem LON-Integrationstool die Adressdaten der Konfigurationsparameter im Gerät zur Verfügung.

### **Konfigurationsparameter Node Object:**

#### **SCPTlocation**

SCPT Index: 17, SNVT\_str\_asc

Funktion: Zusätzliche Eingabemöglichkeit um Informationen zur Standortkennung im Gerät speichern zu können.

#### **UCPTdeviceType**

UCPT Index: 42, SNVT\_count

Funktion: Die Anpassung der Software erfolgt über diesen Konfigurationsparameter. Gültige Eingabewerte sind für:

Gerätevariante 2T	==>	UCPTdeviceType = 2
Gerätevariante 4T	==>	UCPTdeviceType = 4
Gerätevariante 6T	==>	UCPTdeviceType = 6
Gerätevariante 8T	==>	UCPTdeviceType = 8
Gerätevariante 12T	==>	UCPTdeviceType = 12

Voreingestellter Wert: 8

#### **UCPTdiConfig[0]...[11]**

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

Funktion: Diese Konfigurationsparameter bestimmen die Tasterfunktionen und deren Zuordnung zu den Ausgangsvariablen, sowie die Ansteuerlogik der Rückmelde-LEDs.

UCPTdiConfig[0]	konfiguriert	<b>Taste T1</b>
UCPTdiConfig[1]	konfiguriert	<b>Taste T2</b>
:		
UCPTdiConfig[11]	konfiguriert	<b>Taste T12</b>

Beispiel: Taster **T5** konfiguriert für *Switch-Objekt-0* als *Beleuchtungstaster mit Toggle-Funktion*.  
UCPTdiConfig[4].Byte[0] = 0 und UCPTdiConfig[4].Byte[1] = 2

UCPTdiConfig, Konfiguration der Taster und Rückmelde LEDs zur Beleuchtungs- und Jalousiesteuerung					
Zuordnung Taster - Funktionsblock		Taster - Funktion		LED - Funktion	
Byte[0]	Beschreibung	Byte[1]	Beschreibung	Byte[2]	Beschreibung
0	Switch-Objekt 0	1	gedrückt/nicht gedrückt	1	LED = EIN, wenn Taster gedrückt LED = AUS, wenn Taster nicht gedrückt
1	Switch-Objekt 1	2	Licht Toggle	2	LED = EIN, wenn Beleuchtung = EIN LED = AUS, wenn Beleuchtung = AUS (Auswertung von nvoSwitch und nviSwitchFb)
2	Switch-Objekt 2	3	Licht nur EIN	3	LED = EIN, wenn nvoSetting.function = SET_UP LED = AUS, wenn nvoSetting.function = SET_STOP
3	Switch-Objekt 3	4	Licht nur AUS	4	LED = EIN, wenn nvoSetting.function = SET_DOWN LED = AUS, wenn nvoSetting.function = SET_STOP
4	Switch-Objekt 4	5	Licht Toggle mit Dimmen Einschaltwert = Max-Wert	5	LED = EIN, wenn nvoSwitch.state = -1 LED = AUS, wenn nvoSwitch.state ≠ -1
5	Switch-Objekt 5	6	Licht Toggle mit Dimmen Einschaltwert = letzter Ein-Wert		
6	Switch-Objekt 6	7	Licht nur Heller mit Dimmen Einschaltwert = Max-Wert		
7	Switch-Objekt 7	8	Licht nur Heller mit Dimmen Einschaltwert = letzter Ein-Wert		
		9	Licht nur dunkler mit Dimmen		
		0A	Jalousie AUF		
		0B	Jalousie AB		
		0C	Befehl Automatik (= 0.0 -1)		

UCPTdiConfig, Konfiguration der Taster und Rückmelde LEDs zur Szenensteuerung					
Zuordnung Taster - Funktionsblock		Taster - Funktion		LED - Funktion	
Byte[0]	Beschreibung	Byte[1]	Beschreibung	Byte[2]	Beschreibung
8	Scene Panel	01-FE	Szenen-Nummer	1	LED = EIN, wenn Taster gedrückt LED = AUS, wenn Taster nicht gedrückt
				2	LED = EIN, wenn Taster gedrückt oder wenn .scene_number = Byte[1] LED = AUS, wenn .scene_number ≠ Byte[1] (Auswertung von nvoScScene und nviScSceneFb)

UCPTdiConfig, Konfiguration der Taster und Rückmelde LEDs zur Steuerung der Raumbelugung					
Zuordnung Taster - Funktionsblock		Taster - Funktion		LED - Funktion	
Byte[0]	Beschreibung	Byte[1]	Beschreibung	Byte[2]	Beschreibung
9	Occupancy Sensor	1	OCCUPIED, Überstundenfunktion	1	LED = EIN, wenn Taster gedrückt LED = AUS, wenn Taster nicht gedrückt
		2	Toggle Occupied/Unoccupied	2	LED = EIN, wenn nvoOccup = OC_OCCUPIED LED = AUS, wenn nvoOccup ≠ OC_OCCUPIED
		3	nur OCCUPIED	3	LED = EIN, wenn nvoEffectOccup = OC_OCCUPIED LED = AUS, wenn nvoEffectOccup ≠ OC_OCCUPIED
		4	nur UNOCCUPIED		

UCPTdiConfig, Konfiguration der Taster und Rückmelde LEDs zur Klimasteuerung					
Zuordnung Taster - Funktionsblock		Taster - Funktion		LED - Funktion	
Byte[0]	Beschreibung	Byte[1]	Beschreibung	Byte[2]	Beschreibung
0A	Space Comfort Control Command Module	1	Solltemperatur Plus	1	LED = EIN, wenn Taster gedrückt LED = AUS, wenn Taster nicht gedrückt
		2	Solltemperatur Minus	2	LED = EIN, wenn Taster gedrückt oder wenn nvoFanSpeed.state = -1 LED = AUS, wenn nvoFanSpeed.state ≠ -1
		3	Lüfterstufe Plus mit "AUTO"		
		4	Lüfterstufe Minus mit "AUTO"		
		5	Lüfterstufe Plus ohne "AUTO"		
		6	Lüfterstufe Minus ohne "AUTO"		
		7	Lüfterstufe "nur AUTO"		

Hinweis: Die Freigabe der Bedientasten zur Klimasteuerung ist abhängig von der aktuellen Raumbelugung (siehe Space Comfort Control Command Module).  
Nur bei nvoEffectOccup = OC\_OCCUPIED ist eine Verstellung der Werte möglich.

### UCPTtimeConfig

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

Funktion: Mit UCPTtimeConfig lässt sich die Darstellungsform von Wochentag, Datum und Uhrzeit konfigurieren.

!!Bitte Beachten: Nach Modulreset ist die Anzeige für Datum und Uhrzeit ausgeblendet. Erst nach einem empfangenen Telegramm über die Eingangsvariable *nviTimeSet* wird die Uhrzeit eingeblendet. Die Uhrzeit sollte einmal pro Tag über eine LON-Systemuhr synchronisiert werden.

UCPTtimeConfig.Byte[0] = 0	==>	<b>Wochentag</b> ausblenden
UCPTtimeConfig.Byte[0] = 1	==>	<b>Wochentag</b> englisch
UCPTtimeConfig.Byte[0] = 2	==>	<b>Wochentag</b> deutsch
UCPTtimeConfig.Byte[1] = 0	==>	<b>Datum</b> ausblenden
UCPTtimeConfig.Byte[1] = 1	==>	<b>Datum</b> englisch (JJ.MM.TT)
UCPTtimeConfig.Byte[1] = 2	==>	<b>Datum</b> deutsch (TT.MM.JJ)
UCPTtimeConfig.Byte[2] = 0	==>	<b>Uhrzeit</b> ausblenden
UCPTtimeConfig.Byte[2] = 1	==>	<b>Uhrzeit</b> mit Sekunde
UCPTtimeConfig.Byte[2] = 2	==>	<b>Uhrzeit</b> ohne Sekunde
UCPTtimeConfig.Byte[3] = 0	==>	<b>Uhrzeit</b> 24-Stunden-Modus
UCPTtimeConfig.Byte[3] = 1	==>	<b>Uhrzeit</b> 12-Stunden-Modus

Voreingestellte Werte: .Byte[0] = 2, .Byte[1] = 2, .Byte[2] = 2, .Byte[3] = 0

### UCPTdisplConfig

UCPT Index: 46, SNVT\_state

Funktion: Mit UCPTdisplConfig lässt sich die Darstellungsform der Messwerte konfigurieren. Werden mehrere Messwerte angezeigt, schaltet die Anzeige im Zeitintervall *UCPTdisplayTime* zwischen den Messwerten hin und her.

UCPTdisplConfig.bit[0] = 1*	==>	<b>Raumtemperatur</b> anzeigen
UCPTdisplConfig.bit[0] = 0	==>	<b>Raumtemperatur</b> nicht anzeigen
UCPTdisplConfig.bit[1] = 1	==>	<b>Außentemperatur</b> anzeigen
UCPTdisplConfig.bit[1] = 0*	==>	<b>Außentemperatur</b> nicht anzeigen
UCPTdisplConfig.bit[2] = 1	==>	<b>Solltemperatur effektiv</b> anzeigen
UCPTdisplConfig.bit[2] = 0*	==>	<b>Solltemperatur effektiv</b> nicht anzeigen
UCPTdisplConfig.bit[3] = 1	==>	<b>Sollwertoffset</b> anzeigen
UCPTdisplConfig.bit[3] = 0*	==>	<b>Sollwertoffset</b> nicht anzeigen
<small>Wenn bit2 = 1 und bit3 = 1, dann wird die effektive Solltemperatur angezeigt</small>		
UCPTdisplConfig.bit[4] = 1*	==>	<b>°C</b> ist Maßeinheit für die Temperaturanzeige
UCPTdisplConfig.bit[4] = 0	==>	<b>°F</b> ist Maßeinheit für die Temperaturanzeige
UCPTdisplConfig.bit[5] = 1*	==>	<b>Nachkommastelle für Temperatur</b> anzeigen
UCPTdisplConfig.bit[5] = 0	==>	<b>Nachkommastelle für Temperatur</b> nicht anzeigen
UCPTdisplConfig.bit[6] = 1	==>	<b>Rel. Feuchte</b> der Raumluft anzeigen
UCPTdisplConfig.bit[6] = 0*	==>	<b>Rel. Feuchte</b> der Raumluft nicht anzeigen
UCPTdisplConfig.bit[7] = 1	==>	<b>Rel. Feuchte</b> der Außenluft anzeigen
UCPTdisplConfig.bit[7] = 0*	==>	<b>Rel. Feuchte</b> der Außenluft nicht anzeigen
UCPTdisplConfig.bit[8] = 1*	==>	<b>Nachkommastelle für rel. Feuchte</b> anzeigen
UCPTdisplConfig.bit[8] = 0	==>	<b>Nachkommastelle für rel. Feuchte</b> nicht anzeigen
UCPTdisplConfig.bit[9] = 1*	==>	<b>Symbole Raumbelegung</b> anzeigen (Status von nvoEffectOccup)
UCPTdisplConfig.bit[9] = 0	==>	<b>Symbole Raumbelegung</b> nicht anzeigen
UCPTdisplConfig.bit[10] = 1	==>	<b>Symbole Lüfter</b> anzeigen
UCPTdisplConfig.bit[10] = 0*	==>	<b>Symbole Lüfter</b> nicht anzeigen
UCPTdisplConfig.bit[11] = 1	==>	<b>Anzeige Sollwertoffset</b> bei Sollwertverstellung
UCPTdisplConfig.bit[11] = 0*	==>	<b>Anzeige Solltemperatur effektiv</b> bei Sollwertverstellung
UCPTdisplConfig.bit[12] = 1	==>	<b>Anzeige Sollwert</b> bei erstem Tastendruck
UCPTdisplConfig.bit[12] = 0*	==>	<b>Verstellung Sollwert</b> bei erstem Tastendruck
UCPTdisplConfig.bit[13] = 1	==>	<b>nviTempSetpoint</b> ist effektiver Sollwert
UCPTdisplConfig.bit[13] = 0*	==>	<b>nviTempSetpoint</b> ist Basis-Sollwert

\* = Voreingestellte Werte

#### **UCPTdisplayTime**

UCPT Index: 16, SNVT\_time\_sec

Funktion: Dieser Konfigurationsparameter definiert die Zeitdauer einer Anzeigemeldung bzw. das Aktualisierungsintervall der Anzeige. (Voreingestellter Wert: 5,0 sec)

#### **UCPTbacklightMin[0],[1]**

UCPT Index: 48, SNVT\_lev\_cont

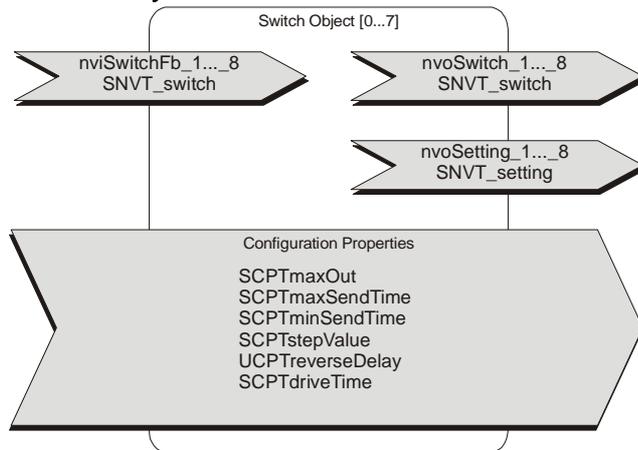
Funktion: Im Ruhezustand, d.h. nach Modulreset oder wenn für länger als 15s kein Tastendruck erfolgt, bestimmt der Wert *UCPTbacklightMin[0]* die Helligkeit der Hintergrundbeleuchtung des LCD. Der Wert *UCPTbacklightMin[1]* bestimmt die Helligkeit des Beschriftungsfeldes.

#### **UCPTbacklightMax[0],[1]**

UCPT Index: 47, SNVT\_lev\_cont

Funktion: Wird der Ruhezustand durch Betätigung einer Taste verlassen, dann bestimmt der Wert *UCPTbacklightMax[0]* die Helligkeit der Hintergrundbeleuchtung des LCD. Der Wert *UCPTbacklightMax[1]* bestimmt die Helligkeit des Beschriftungsfeldes.

## Switch Object



Acht identische Switch-Objekte für die Beleuchtungs- und Jalousiesteuerung. Sowohl die Zuordnung der Tasten zu den einzelnen Objekten als auch die Funktionsfestlegung erfolgt wie zuvor beschrieben mit den Konfigurationsparametern UCPTdiConfig[0]...[11] im Node Object. Mögliche Funktionen sind Beleuchtungssteuerung EIN/AUS, Beleuchtungssteuerung EIN/AUS mit Dimmen oder Jalousie AUF/AB mit Automatiklauf.

### Eingangsvariablen Switch Object:

#### nviSwitchFb\_1...8

SNVT Typ: SNVT\_switch, Index 95

Funktion: Eingangsvariablen für den aktuellen Zustand der mit nvoSwitch\_1...8 angesteuerten Beleuchtungsgruppen.

### Ausgangsvariablen Switch Object:

#### nvoSwitch\_1...8, nvoSetting\_1...8

SNVT Typ: SNVT\_switch, Index 95; SNVT\_setting, Index 117

Funktion: Ausgangsvariablen zur Ansteuerung von Beleuchtungsgruppen und von Beleuchtungs- und Jalousiecontrollern. Die Ausgabewerte sind von den Funktionseinstellungen abhängig. Mit UCPTdiConfig.Byte[0] = 0 - 7 werden Taster einem Switch-Object zugeordnet. UCPTdiConfig.Byte[1] = 0 - 0C hex weist diesen Tastern dann eine Funktion zu. Es können auch mehrere Tasten einem Switch-Object zugeordnet werden, z.B. T5 zum Ausschalten und T6 zum Einschalten der Beleuchtung.

#### UCPTdiConfig.Byte[1] = 01<sub>hex</sub>, Taste gedrückt/ nicht gedrückt

Taste gedrückt	nvoSwitch.value	= SCPTmaxOut
	nvoSwitch.state	= 1
Taste nicht gedrückt	nvoSetting.function	= SET_ON;
	nvoSetting.setting	= SCPTmaxOut;
Taste nicht gedrückt	nvoSwitch.value	= 0
	nvoSwitch.state	= 0
Taste nicht gedrückt	nvoSetting.function	= SET_OFF;
	nvoSetting.setting	= 0;

#### UCPTdiConfig.Byte[1] = 02<sub>hex</sub>, Beleuchtung Toggle

Jede Tastbetätigung führt zum Umschalten der Beleuchtung, d.h. zwischen EIN und AUS

Beleuchtung EIN	nvoSwitch.value	= SCPTmaxOut
	nvoSwitch.state	= 1
Beleuchtung EIN	nvoSetting.function	= SET_ON;
	nvoSetting.setting	= SCPTmaxOut;
Beleuchtung AUS	nvoSwitch.value	= 0
	nvoSwitch.state	= 0
Beleuchtung AUS	nvoSetting.function	= SET_OFF;
	nvoSetting.setting	= 0;

**UCPTdiConfig.Byte[1] = 03<sub>hex</sub>, Beleuchtung EIN**

Jede Tastbetätigung führt zum Einschalten der Beleuchtung

Beleuchtung EIN	nvoSwitch.value	= SCPTmaxOut
	nvoSwitch.state	= 1
	nvoSetting.function	= SET_ON;
	nvoSetting.setting	= SCPTmaxOut;

**UCPTdiConfig.Byte[1] = 04<sub>hex</sub>, Beleuchtung AUS**

Jede Tastbetätigung führt zum Ausschalten der Beleuchtung

Beleuchtung AUS	nvoSwitch.value	= 0
	nvoSwitch.state	= 0
	nvoSetting.function	= SET_OFF;
	nvoSetting.setting	= 0;

**UCPTdiConfig.Byte[1] = 05<sub>hex</sub>, Beleuchtung Toggle mit Dimmen, Einschaltwert = max. Wert**

Kurze Tastbetätigungen (< 1 s) führen zum Umschalten des aktuellen Beleuchtungszustandes, wobei der value - Einschaltwert immer SCPTmaxOut beträgt. Mit längeren Tastbetätigungen (> 1 s) wird die Dimm-Funktion aktiviert, d.h. ausgehend vom aktuellen Beleuchtungszustand wird der .value-Wert der Switch-Variablen in Prozentschritten von UCPTstepValue erhöht oder verringert und zwar so lange wie die Taste gedrückt wird.

Beleuchtung auf Maximalwert	nvoSwitch.value	= SCPTmaxOut
	nvoSwitch.state	= 1
Beleuchtung auf 50%	nvoSwitch.value	= 50,0
	nvoSwitch.state	= 1
Beleuchtung AUS	nvoSwitch.value	= 0
	nvoSwitch.state	= 0

Die Variablen nvoSetting werden in dieser Einstellung nicht verändert.

**UCPTdiConfig.Byte[1] = 06<sub>hex</sub>, Beleuchtung Toggle mit Dimmen, Einschaltwert = letzter Ein-Wert**

Funktion wie bei 05hex, nur mit dem Unterschied, dass beim Einschalten der Beleuchtung nicht der Wert SCPTmaxOut, sondern der letzte Einschaltwert eingenommen wird. Der kleinste Einschaltwert ist hierbei auf 20% begrenzt.

**UCPTdiConfig.Byte[1] = 07<sub>hex</sub>, Beleuchtung EIN mit Heller-Dimmen, Einschaltwert = max. Wert**

Ist die Beleuchtung ausgeschaltet, dann führt eine Tastbetätigung (< 1 s) zum sofortigen Einschalten der Beleuchtung. Ist die Beleuchtung bereits eingeschaltet und gedimmt, so kann diese mit einem weiteren Tastendruck (< 1 s) unabhängig vom aktuellen Beleuchtungszustand auf den Maximalwert SCPTmaxOut geschaltet werden. Mit längeren Tastbetätigungen (> 1 s) wird die Funktion „Heller-Dimmen“ aktiviert, d.h. ausgehend vom aktuellen Beleuchtungszustand wird der .value - Wert der Switch-Variablen in Prozent - Schritten von UCPTstepValue erhöht und zwar so lange bis der Maximalwert SCPTmaxOut erreicht wird. Das Sendeintervall im Modus Dimmen wird mit SCPTminSendTime eingestellt und beträgt voreingestellt ca. 300ms.

Beleuchtung Einschalten	nvoSwitch.value	= SCPTmaxOut
	nvoSwitch.state	= 1
	nvoSetting.function	= SET_ON;
	nvoSetting.setting	= SCPTmaxOut;
Beleuchtung Heller-Dimmen	nvoSwitch.value	= letzter Wert + UCPTstepValue
	nvoSwitch.state	= 1
	nvoSetting.function	= SET_UP;
	nvoSetting.setting	= UCPTstepValue;

**UCPTdiConfig.Byte[1] = 08<sub>hex</sub>, Beleuchtung EIN mit Heller-Dimmen, Einschaltwert = letzter Ein-Wert**

Funktion wie bei 07hex, nur mit dem Unterschied, dass beim Einschalten der Beleuchtung nicht der Wert SCPTmaxOut, sondern der letzte Einschaltwert eingenommen wird. Der kleinste Einschaltwert ist hierbei auf 20% begrenzt.

**UCPTdiConfig.Byte[1] = 09<sub>hex</sub>, Beleuchtung AUS mit Dunkler-Dimmen**

Ist die Beleuchtung eingeschaltet, dann führt eine kurze Tastbetätigung (< 1 s) zum sofortigen ausschalten der Beleuchtung. Mit längeren Tastbetätigungen (> 1 s) wird die Funktion „Dunkler-Dimmen“ aktiviert, d.h. ausgehend vom aktuellen Beleuchtungszustand wird der .value - Wert der Switch-Variablen in Prozent - Schritten von UCPTstepValue verringert und zwar so lange bis der Wert 0 erreicht wird. Das Sendeintervall im Modus Dimmen wird mit SCPTminSendTime eingestellt und beträgt voreingestellt ca. 300ms.

Beleuchtung Ausschalten	nvoSwitch.value	= 0
	nvoSwitch.state	= 0
	nvoSetting.function	= SET_OFF;
	nvoSetting.setting	= 0;
Beleuchtung Dunkler-Dimmen	nvoSwitch.value	= letzter Wert - UCPTstepValue
	nvoSwitch.state	= 1
	nvoSetting.function	= SET_DOWN;
	nvoSetting.setting	= UCPTstepValue;

**UCPTdiConfig.Byte[1] = 0A<sub>hex</sub>, Jalousie AUF**

Im Konfigurationsmodus „Jalousie AUF“ werden nur die nvoSetting-Variablen verändert und gesendet. Kurze Tastbetätigungen (< 2 s) dienen zur Feineinstellung der Lamellen. Eine lange Tastbetätigung (> 2 s) startet den Automatiklauf und steuert die Jalousie für die Zeit SCPTdriveTime dauerhaft in Richtung öffnen an. Der Automatiklauf kann durch erneute Tastbetätigung angehalten werden.

Jalousie öffnen	nvoSetting.function	= SET_UP;
Jalousie anhalten	nvoSetting.function	= SET_STOP;

**UCPTdiConfig.Byte[1] = 0B<sub>hex</sub>, Jalousie AB**

Im Konfigurationsmodus „Jalousie AB“ werden nur die nvoSetting-Variablen verändert und gesendet. Kurze Tastbetätigungen (< 2 s) dienen zur Feineinstellung der Lamellen. Eine lange Tastbetätigung (> 2 s) startet den Automatiklauf und steuert die Jalousie für die Zeit SCPTdriveTime dauerhaft in Richtung schließen an. Der Automatiklauf kann durch erneute Tastbetätigung angehalten werden.

Jalousie schließen	nvoSetting.function	= SET_DOWN;
Jalousie anhalten	nvoSetting.function	= SET_STOP;

**UCPTdiConfig.Byte[1] = 0C<sub>hex</sub>, Befehl Automatik**

Die Betätigung einer „Automatik-Taste“ schaltet die Variablen nvoSwitch auf den Wert 0,0 -1. Damit kann z.B. ein Beleuchtungsregler durch ext. Übersteuerung wieder in den Automatikmodus gebracht werden.

**Konfigurationsparameter Switch Object:**

**SCPTmaxOut**

SCPT Index: 93, SNVT\_lev\_cont

Funktion: Dieser Konfigurationsparameter bestimmt den maximalen Ausgabewert der Variablen nvoSwitch.value.

#### **SCPTmaxSendTime**

SCPT Index: 49, SNVT\_time\_sec

Funktion: Heartbeatfunktion. Dieser Konfigurationsparameter legt die Intervallzeit fest, nach der die Ausgangsvariablen gesendet werden. Mit Eingabewerten = 0 wird die Heartbeatfunktion deaktiviert. (Voreingestellter Wert: 0,0 s)

#### **SCPTminSendTime**

SCPT Index: 52, SNVT\_time\_sec

Funktion: Dieser Konfigurationsparameter legt das Sendeintervall der Ausgangsvariablen im Modus Dimmen fest. Mit Eingabewerten = 0 wird die Funktion deaktiviert. (Voreingestellter Wert: 0,3 s)

#### **SCPTstepValue**

SCPT Index: 92, SNVT\_lev\_cont

Funktion: Dieser Konfigurationsparameter definiert die Schrittweite der Variablen nvoSwitch.value im Modus Dimmen.

#### **UCPTreverseDelay**

UCPT Index: 14, SNVT\_count

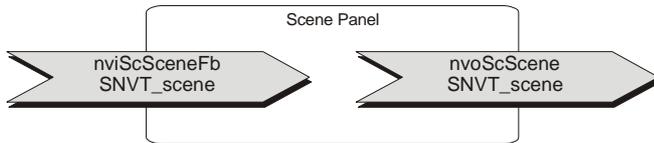
Funktion: Dieser Konfigurationsparameter definiert die Umschaltverzögerung bei Drehrichtungsumkehr der Jalousiemotoren in ms. Damit wird ein Wechselbefehl z.B. von nvoSetting = SET\_UP auf nvoSetting = SET\_DOWN verzögert ausgegeben. (Voreingestellter Wert: 500 ms)

#### **SCPTdriveTime**

UCPT Index: 45, SNVT\_time\_sec

Funktion: Dieser Konfigurationsparameter definiert die maximale Einschaltzeit der Jalousiemotoren im Automatiklauf. (Voreingestellter Wert: 100,0 s)

## Scene Panel



### Eingangsvariablen Scene Panel:

#### nviScSceneFb

SNVT Typ: SNVT\_scene, Index 115

Funktion: Eingangsvariable mit der aktuellen Beleuchtungsszene im Raum. Bei Empfang eines neuen Wertes werden die Konfigurationen aller Taster und deren LEDs überprüft. Bei entsprechender Einstellung wird die aktuelle Szene dann durch die entsprechende LED angezeigt.

### Ausgangsvariablen Scene Panel:

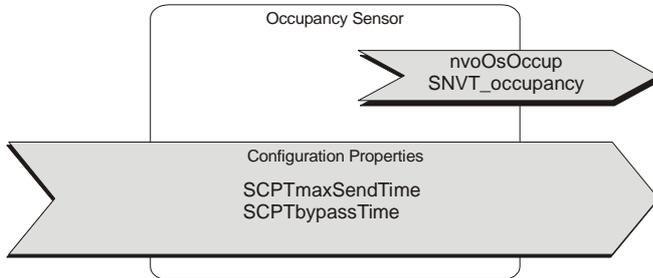
#### nvoScScene

SNVT Typ: SNVT\_scene, Index 115

Funktion: Ausgangsvariable zur Ansteuerung eines Szene-Controllers. Die Ausgabewerte sind von den Funktionseinstellungen abhängig. Mit UCPTdiConfig.Byte[0] = 8 werden Taster dem Scene-Object zugeordnet. UCPTdiConfig.Byte[1] = 01<sub>hex</sub> - FE<sub>hex</sub> weist diesen Tastern dann eine Szenennummer zu, die bei kurzen Tastbetätigungen mit SC\_RECALL aufgerufen wird. Bei langen Tastbetätigungen (> 2 s) wird die Szene mit SC\_LEARN neu gelernt. Zur Bestätigung einer neu gelernten Szene blinkt die Taster-Rückmelde-LED kurz. Es können auch mehrere Tasten dem Scene Panel zugeordnet werden, z.B. Taster T5 zum Aufruf von Szene 1 und Taster T6 zum Aufruf von Szene 2.

## Occupancy Sensor Object

### AusgangsvARIABLEN Occupancy Sensor Object:



### nvoOsOccup

SNVT Typ: SNVT\_occupancy, Index 109

Funktion: AusgangsvARIABLE zur Präsenzmeldung im Raum. Die Ausgabewerte sind von den Funktionseinstellungen abhängig. Mit UCPTdiConfig.Byte[0] = 9 werden Taster dem Occupancy-Sensor-Object zugeordnet. UCPTdiConfig.Byte[1] = 01<sub>hex</sub> - 04<sub>hex</sub> weist diesen Tastern dann eine Funktion zu. Es können auch mehrere Tasten einem Occupancy-Sensor-Object zugeordnet werden. nvoOsOccup kann zur lokalen Präsenzmeldung auf die Eingangsvariable nviOccSensor gebunden werden.

#### UCPTdiConfig.Byte[1] = 01<sub>hex</sub> Präsenztaste mit Überstundenfunktion

Durch Tastbetätigung erhält die AusgangsvARIABLE nvoOsOccup den Wert OC\_OCCUPIED und die Nachlaufzeit wird gestartet. Nach Ablauf der Verzögerungszeit SCPTbypassTime wird die AusgangsvARIABLE wieder auf den Wert UNOCCUPIED zurückgesetzt. Jede Tastbetätigung startet die Nachlaufzeit von neuem.

#### UCPTdiConfig.Byte[1] = 02<sub>hex</sub> Präsenztaste mit Toggle-Funktion

Durch Tastbetätigung wird die AusgangsvARIABLE nvoOsOccup zwischen den Werten OC\_OCCUPIED und OC\_UNOCCUPIED hin und her geschaltet.

#### UCPTdiConfig.Byte[1] = 03<sub>hex</sub> Präsenztaste OCCUPIED

Durch Tastbetätigung erhält die AusgangsvARIABLE nvoOsOccup den Wert OC\_OCCUPIED.

#### UCPTdiConfig.Byte[1] = 04<sub>hex</sub> Präsenztaste UNOCCUPIED

Durch Tastbetätigung erhält die AusgangsvARIABLE nvoOsOccup den Wert OC\_UNOCCUPIED.

### Konfigurationsparameter Occupancy Sensor Object:

#### SCPTbypassTime

SCPT Index: 34, SNVT\_time\_min

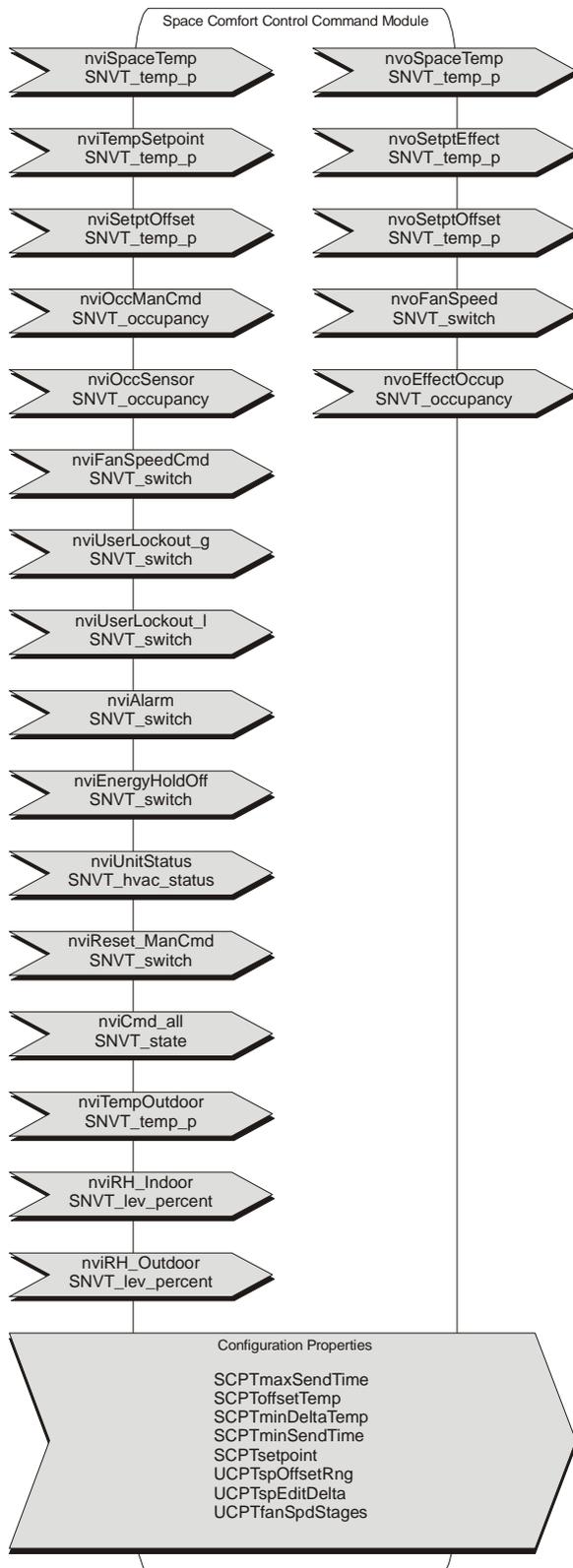
Funktion: Verzögerungszeit in Minuten. Nach Ablauf von SCPTbypassTime wird die AusgangsvARIABLE nvoOsOccup auf OC\_UNOCCUPIED zurückgesetzt. (Voreingestellter Wert: 90 min)

#### SCPTmaxSendTime

SCPT Index: 49, SNVT\_time\_sec

Funktion: Heartbeatfunktion. Legt die Intervallzeit fest, nach der die AusgangsvARIABLEN gesendet werden. Mit Eingabewerten = 0 wird die Heartbeatfunktion deaktiviert. (Voreingestellter Wert: 120 s)

## Space Comfort Control Command Module:



**Temperaturerfassung:** Die Temperaturerfassung erfolgt entweder mit dem internen Temperaturfühler oder über die Eingangsvariable *nviSpaceTemp* mit externem LON-Fühler.

Für eine nachträgliche Kalibrierung des internen Sensors steht der Konfigurationsparameter *SCPToffsetTemp* zur Verfügung.

**Messwertanzeigen:** Zusätzlich zur gemessenen Raumtemperatur besteht die Möglichkeit die Werte der Eingangsvariablen *nviTempOutdoor*, *nviRH\_Indoor* und *nviRH\_Outdoor* in der Anzeige darzustellen. Die Konfiguration der Anzeige erfolgt im Node Object mit den Parametern *UCPTdisplConfig* und *UCPTdisplayTime*. Soll mehr als ein Messwert angezeigt werden, dann schaltet die Anzeige im Zeitintervall zwischen den Messwerten hin und her.

**Sollwertvorgabe:** Der effektive Sollwert (Basissollwert) *nvoSetptEffect* errechnet sich in Abhängigkeit der Eingangsvariablen zur Raumebelegung (*nviOccManCmd* und *nviOccSensor*) aus den Sollwertvorgaben über *SCPTsetpoint* bzw. *nviTempSetpoint* und dem Offsetwert *nvoSetptOffset* (siehe Tabelle 1).

Soll der Sollwert über mehrere Geräte in einen Raum vorgegeben werden, dann besteht die Möglichkeit die Sollwerteinstellungen mit *nviSetptOffset* und *nvoSetptOffset* zu synchronisieren. Zur Auswertung der internen Bewegungsmeldung muss die Ausgangsvariable *nvoOcOccup* des „Occupancy Sensors“ mit der Eingangsvariablen *nviOccSensor* verbunden werden.

Der Offsetwert *nvoSetptOffset* lässt sich über entsprechend parametrisierte Bedientasten in einem Bereich *UCPTspOffsetRng* anheben bzw. absenken.

**Lüfterverstellung:** Der Ausgabewert für die Lüfterstufen *nvoFanSpeed* kann über entsprechend konfigurierte Taster verstellt werden. Die manuellen Änderungen können mit *nviFanSpeedCmd* übersteuert werden. Sowohl Anzeige als auch Ausgangsvariable übernehmen dann den extern vorgegebenen Wert.

## Eingangsvariablen Space Comfort Control Command Module:

### nviSpaceTemp

SNVT Typ: SNVT\_temp\_p, Index 105

Funktion: Eingangsvariable zum Anschluss eines externen LON-Temperaturfühlers. Der ext. Wert wird übernommen, wenn der Initialisierungswert 0x7FFF (=327,67 °C) nach Reset durch ein NV-Update verändert wurde. Solange der Initialisierungswert nach Reset nicht verändert wird, bleibt der interne Temperaturfühler aktiv!

### nviTempSetpoint

SNVT Typ: SNVT\_temp\_p, Index 105

Funktion: Eingangsvariable zur Vorgabe der Basis-Sollwerttemperatur. Es ist nicht zwingend erforderlich diese Netzwerkvariable mit einem übergeordneten Knoten zu binden. Wenn für nviSetpoint kein Update erfolgt, dann bleibt der Initialisierungswert 0x7FFF (=327,67°C) erhalten und es wird zur Berechnung des effektiven Sollwertes (Basis-Sollwert + Offset) der Wert des Konfigurationsparameters **SCPTsetpoint** herangezogen. Erhält nviSetpoint ein Update mit einem gültigen Sollwert, dann wird der effektive Sollwert mit dem Wert der Eingangsvariablen berechnet.

### nviSetptOffset

SNVT Typ: SNVT\_temp\_p, Index 105

Funktion: Eingangsvariable zur externen Vorgabe für die Sollwertkorrektur. Wird ein neuer Wert empfangen (*nviSetptOffset* ungleich *nvoSetptOffset*), dann übernimmt *nvoSetptOffset* den Wert von *nviSetptOffset*. Damit besteht die Möglichkeit mehrere Geräte in einen Raum zu synchronisieren.

### nviOccManCmd und nviOccSensor

SNVT Typ: SNVT\_occupancy, Index 109

Funktion: Eingangsvariablen zur Vorgabe der Raumbelugung. Die aktuelle Raumbelugung bestimmt die Größen der Regelparameter „effektiver Sollwert“ und „Neutrale Zone“ und damit die Sollwerte für Heizen und Kühlen (siehe Tabelle 1). Initialisierungswert für beide Variablen: OC\_NUL

nviOccManCmd: Vorgabe über GLT:

OC\_OCCUPIED, OC\_BYPASS, OC\_STANDBY, OC\_UNOCCUPIED

nviOccSensor: Präsenzmeldung im Raum:

OC\_OCCUPIED, OC\_UNOCCUPIED

nviOccManCmd	nviOccSensor	>>>	nvoEffectOccup	nvoSetptEffect
OC_NUL	OC_NUL	>>>	OCCUPIED	SCPTsetpoint + nvoSetptOffset  oder nviTempSetpoint + nvoSetptOffset
OC_OCCUPIED	****	>>>		
OC_BYPASS	OC_OCCUPIED	>>>		
OC_NUL	OC_OCCUPIED	>>>		
OC_STANDBY	****	>>>	STANDBY	SCPTsetpoint + nvoSetptOffset  oder nviTempSetpoint + nvoSetptOffset
OC_NUL	OC_UNOCCUPIED	>>>	UNOCCUPIED	SCPTsetpoint  oder nviTempSetpoint
OC_BYPASS	OC_UNOCCUPIED	>>>		
OC_UNOCCUPIED	****	>>>		

**Tabelle 1: Effektiver Sollwert in Abhängigkeit der Raumbelegung**

**nviFanSpeedCmd**

SNVT Typ: SNVT\_switch, Index 95

Funktion: Eingangsvariable zur externen Vorgabe der angezeigten und mit *nvoFanSpeed* ausgegebenen Lüftereinstellung. Der Wertebereich entspricht dem der Ausgangsvariablen *nvoFanSpeed*.

**nviUserLockout\_g**

SNVT Typ: SNVT\_switch, Index 95

Funktion: Eingangsvariable zur Sperrung der Bedientasten zur Szenensteuerung, Raumbelegung und Klimasteuerung.

nviUserLockout\_g = 100.0 1 ==> Tasten sind gesperrt  
nviUserLockout\_g = 0.0 0 ==> Tasten sind freigegeben

### nviUserLockout\_I

SNVT Typ: SNVT\_switch, Index 95

Funktion: Eingangsvariable zur Sperrung der Bedientasten zur Beleuchtungs- und Jalousiesteuerung.

nviUserLockout\_I = 100.0 1 ==> Tasten sind gesperrt  
nviUserLockout\_I = 0.0 0 ==> Tasten sind freigegeben

### nviAlarm

SNVT Typ: SNVT\_switch, Index 95

Funktion: Die Eingangsvariable steuert das LCD-Symbol „Störung“:

nviAlarm = 100.0 1 ==> Symbol „Störung“ ein  
nviAlarm = 50.0 1 ==> Symbol „Störung“ blinkt  
nviAlarm = 0.0 0 ==> Symbol „Störung“ aus

### nviEnergyHoldOff

SNVT Typ: SNVT\_switch, Index 95

Funktion: Die Eingangsvariable steuert das LCD-Symbol „Fenster“:

nviEnergyHoldOff = 100.0 1 ==> Symbol „Fenster“ ein  
nviEnergyHoldOff = 0.0 1 ==> Symbol „Fenster“ aus

### nviUnitStatus

SNVT Typ: SNVT\_hvac\_status, Index 112

Funktion: Die Eingangsvariable steuert die LCD-Symbole „Heizen“ und „Kühlen“:

nviUnitStatus.heat\_output\_primary > 0 ==> Symbol „Heizen“ EIN  
nviUnitStatus.heat\_output\_primary = 0 ==> Symbol „Heizen“ AUS  
nviUnitStatus.cool\_output > 0 ==> Symbol „Kühlen“ EIN  
nviUnitStatus.cool\_output = 0 ==> Symbol „Kühlen“ AUS

### nviReset\_ManCmd

SNVT Typ: SNVT\_switch, Index 95

Funktion: Ein Update der Variablen von 0.0 0 auf 100.0 1 führt zum Zurücksetzen der Benutzereinstellungen.

Sollwertoffset Temperatur: nvoSetptEffect, nvoSetptOffset = 0  
Raumbelegung: nvoOsOccup = OC\_UNOCCUPIED  
Lüfterstufe: nvoFanSpeed = 0,0 -1 (AUTO)

### nviCmd\_all

SNVT Typ: SNVT\_state, Index 83

Funktion:

nviCmd\_all.bit[0] = 1 ==> siehe **nviUserLockout\_g** = 100.0 1  
nviCmd\_all.bit[0] = 0 ==> siehe **nviUserLockout\_g** = 0.0 0  
nviCmd\_all.bit[1] = 1 ==> siehe **nviUserLockout\_I** = 100.0 1  
nviCmd\_all.bit[1] = 0 ==> siehe **nviUserLockout\_I** = 0.0 0  
nviCmd\_all.bit[2] = 1 ==> siehe **nviReset\_ManCmd** = 100.0 1  
nviCmd\_all.bit[2] = 0 ==> siehe **nviReset\_ManCmd** = 0.0 0  
nviCmd\_all.bit[3] = 1 ==> Symbol **Heizen** einschalten  
nviCmd\_all.bit[3] = 0 ==> Symbol **Heizen** ausschalten  
nviCmd\_all.bit[4] = 1 ==> Symbol **Kühlen** einschalten  
nviCmd\_all.bit[4] = 0 ==> Symbol **Kühlen** ausschalten  
nviCmd\_all.bit[5] = 1 ==> Symbol **Fenster** einschalten  
nviCmd\_all.bit[5] = 0 ==> Symbol **Fenster** ausschalten  
nviCmd\_all.bit[6] = 1 ==> Symbol **Störung** einschalten  
nviCmd\_all.bit[6] = 0 ==> Symbol **Störung** ausschalten

### nviTempOutdoor

SNVT Typ: SNVT\_temp\_p, Index 105

Funktion: Über diese Eingangsvariable kann optional die Außentemperatur vorgegeben werden.

#### **nviRH\_Indoor**

SNVT Typ: SNVT\_lev\_percent, Index 81  
Funktion: Über diese Eingangsvariable kann optional die rel. Feuchte der Raumluft vorgegeben werden.

#### **nviRH\_Outdoor**

SNVT Typ: SNVT\_lev\_percent, Index 81  
Funktion: Über diese Eingangsvariable kann optional die rel. Feuchte der Außenluft vorgegeben werden.

### **Ausgangsvariablen Space Comfort Control Command Module:**

#### **nvoSpaceTemp**

SNVT Typ: SNVT\_temp\_p, Index 105  
Funktion: Ausgangsvariable für den gemessenen Temperaturwert. Messbereich 0 - 50°C, Auflösung 1/100 °C. Die Datenausgabe erfolgt in Abhängigkeit der Konfigurationsparameter SCPTmaxSendTime, SCPTminDeltaTemp und SCPTminSendTime und ca. 1,5s - 4,0s nach Reset.

#### **nvoSetptEffect**

SNVT Typ: SNVT\_temp\_p, Index 105  
Funktion: Ausgangsvariable für den effektiven Sollwert. Der effektive Sollwert berechnet sich über die Vorgaben aus nviTempSetpoint bzw. SCPTsetpoint und der am Gerät eingestellten Sollwertverschiebung (siehe Tab. 1).  
Die Datenausgabe erfolgt nach Veränderung des Wertes durch den Benutzer, in Abhängigkeit der Konfigurationsvariablen SCPTmaxSendTime und ca. 1,5s - 4,0s nach Reset.

#### **nvoSetptOffset**

SNVT Typ: SNVT\_temp\_p, Index 105  
Funktion: Ausgangsvariable für die Sollwertkorrektur, die über die Bedientasten verändert werden kann. Die Datenausgabe erfolgt mit jeder Tastbetätigung.  
Mit UCPTdiConfig.Byte[0] = 0A<sub>hex</sub> werden Taster dem Space-Comfort-Control-Command-Modul zugeordnet.  
UCPTdiConfig.Byte[1] = 01<sub>hex</sub> - 02<sub>hex</sub> weist diesen Tastern dann eine Funktion zur Sollwertverstellung zu.  
Es können auch mehrere Tasten dem Space-Comfort-Controller-Object zugeordnet werden.

#### **UCPTdiConfig.Byte[1] = 01<sub>hex</sub> Sollwertoffset erhöhen**

Durch Tastbetätigung wird die Ausgangsvariable nvoSetptOffset um den Wert UCPTspEditDelta erhöht. Der Verstellbereich wird durch UCPTspOffsetRng festgelegt.

#### **UCPTdiConfig.Byte[1] = 02<sub>hex</sub> Sollwertoffset verkleinern**

Durch Tastbetätigung wird die Ausgangsvariable nvoSetptOffset um den Wert UCPTspEditDelta verkleinert. Der Verstellbereich wird durch UCPTspOffsetRng festgelegt.

!! Die Betätigung der Tasten wird **nur bei nvoEffectOccup = OC\_OCCUPIED** ausgewertet. Ansonsten werden die Tastbetätigungen ignoriert. Die Anzeige im Display zeigt für die Dauer der Sollwertverstellung je nach Konfiguration entweder den effektiven Sollwert nvoEffectSetpt oder den Offsetwert nvoSetptOffset an.

#### **nvoEffectOccup**

SNVT Typ: SNVT\_occupancy, Index 109  
Funktion: Ausgangsvariablen für die effektive Raumbelugung (siehe Tabelle 1).  
Die Datenausgabe erfolgt in Abhängigkeit von SCPTmaxSendTime, bei Wertänderungen und ca. 1,5s - 4,0s nach Reset.

#### **nvoFanSpeed**

SNVT Typ: SNVT\_switch, Index 95  
Funktion: Ausgangsvariablen für die eingestellte Lüftergeschwindigkeit. Die Anzahl der Lüfterstufen ist mit dem Konfigurationsparameter UCPTfanSpdStages einstellbar. Die Anzeige im Display

zeigt die aktuell eingestellt Lüftergeschwindigkeit an und ob der Regler sich im Automatikmodus befindet.

UCPTfanSpdStages = 1			UCPTfanSpdStages = 2			UCPTfanSpdStages = 3		
Lüfterstufe	nvoMultiOut .value	nvoMultiOut .state	Lüfterstufe	nvoMultiOut .value	nvoMultiOut .state	Lüfterstufe	nvoMultiOut .value	nvoMultiOut .state
AUTO	0 %	-1	AUTO	0 %	-1	AUTO	0 %	-1
0	0 %	1	0	0 %	1	0	0 %	1
1	100 %	1	1	50 %	1	1	33,0 %	1
			2	100 %	1	2	66,5 %	1
						3	100 %	1

!!Die Betätigung der Tasten wird **nur bei nvoEffectOccup = OC\_OCCUPIED** ausgewertet. Ansonsten werden die Tastbetätigungen ignoriert.

Mit UCPTdiConfig.Byte[0] = 0A<sub>hex</sub> werden Taster dem Space-Comfort-Controller-Object zugeordnet.

UCPTdiConfig.Byte[1] = 03<sub>hex</sub> - 07<sub>hex</sub> weist diesen Tastern dann eine Funktion zur Lüfterstufenverstellung zu. Es können auch mehrere Tasten dem Space-Comfort-Controller-Object zugeordnet werden.

**UCPTdiConfig.Byte[1] = 03<sub>hex</sub> Lüfterstufe erhöhen mit AUTO**

Durch Tastbetätigung wird die Ausgangsvariable *nvoFanSpeed* erhöht. Eine Tastbetätigung bei max. Lüfterstufe schaltet wieder zurück auf AUTO.

**UCPTdiConfig.Byte[1] = 04<sub>hex</sub> Lüfterstufe verkleinern mit AUTO**

Durch Tastbetätigung wird die Ausgangsvariable *nvoFanSpeed* verkleinert. Eine Tastbetätigung bei AUTO schaltet wieder zurück auf die maximale Stufe.

**UCPTdiConfig.Byte[1] = 05<sub>hex</sub> Lüfterstufe erhöhen ohne AUTO**

Durch Tastbetätigung wird die Ausgangsvariable *nvoFanSpeed* erhöht. Eine Tastbetätigung bei max. Lüfterstufe schaltet wieder zurück auf „0“. Der Wert AUTO wird übersprungen.

**UCPTdiConfig.Byte[1] = 06<sub>hex</sub> Lüfterstufe verkleinern ohne AUTO**

Durch Tastbetätigung wird die Ausgangsvariable *nvoFanSpeed* verkleinert. Eine Tastbetätigung bei „0“ schaltet wieder zurück auf die maximale Stufe. Der Wert AUTO wird übersprungen.

**UCPTdiConfig.Byte[1] = 07<sub>hex</sub> Lüfterstufe nur AUTO**

Durch Tastbetätigung wird die Ausgangsvariable *nvoFanSpeed* auf den Wert 0.0 -1 geschaltet.

## Konfigurationsparameter Space Comfort Control Command Module

### SCPTmaxSendTime

Index: 49, SNVT\_time\_sec

Funktion: Heartbeatfunktion. Dieser Konfigurationsparameter legt die Intervallzeit fest, nach der die Ausgangsvariablen gesendet werden. Mit Eingabewerten = 0 wird die Heartbeatfunktion deaktiviert.  
(Voreingestellter Wert: 300,0 s)

### SCPTminSendTime

Index: 52, SNVT\_time\_sec

Funktion: Legt das kleinste Update-Intervall der Ausgangsvariablen nvoSpaceTemp fest. Ein Update erfolgt nach Ablauf von „SCPTminSendTime“, wenn sich der Temperaturwert der Ausgangsvariablen um mehr als „SCPTminDeltaTemp“ geändert hat. Mit Eingabewerten = 0 wird die Funktion deaktiviert.  
(Voreingestellter Wert: 5,0 sec)

#### **SCPTminDeltaTemp**

Index: 64, SNVT\_temp\_p

Funktion: Wenn sich die Temperatur um den eingestellten Wert „SCPTminDeltaTemp“ verändert, wird der neue Temperaturwert übertragen. Die Funktion ist abhängig von der Einstellung des Parameters „SCPTminSendTime“.  
(Wertebereich  $\geq 0$  °C; Voreingestellter Wert: 0,30 °C)

#### **SCPToffsetTemp**

Index: 70, SNVT\_temp\_p

Funktion: Offset für den Temperaturwert. Mit diesem Parameter ist eine softwareseitige Kalibrierung möglich. Bitte beachten Sie dazu die Hinweise für Wohnraumfühler Unterputz mit Messumformer.

#### **SCPTsetpoint**

Index: 213, SNVT\_temp\_p

Funktion: Konfigurationsparameter zur Sollwertvorgabe (Voreingestellter Wert: 22,00 °C)

#### **UCPTspEditDelta**

Index: 9, SNVT\_temp\_p

Funktion: Konfigurationsparameter zur Definition der Temperatursprünge bei der Sollwertverstellung über die Bedientaster. (Voreingestellter Wert: 0,5K, d.h. mit jeder Tastbetätigung ändert sich der Sollwert um 0,5 K)

#### **UCPTspOffsetRng**

Index: 12, SNVT\_temp\_p

Funktion: Konfigurationsparameter für den Wertebereich der einstellbaren Sollwertkorrektur, d.h. der vorgegebene Sollwert lässt sich um den Wert +/- UCPTspOffsetRng durch den Benutzer verändern.  
(Voreingestellter Wert: 3,0 K)

#### **UCPTfanSpdStages**

Index: 13, SNVT\_count

Funktion: Konfigurationsparameter zur Vorgabe der Lüfterstufen.  
(Voreingestellter Wert: 3 ==> AUS, 33,0 %, 66,5 %, 100,0 %, AUTO)