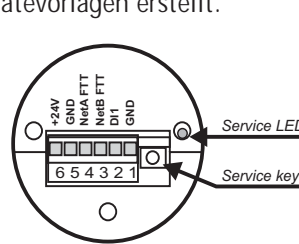
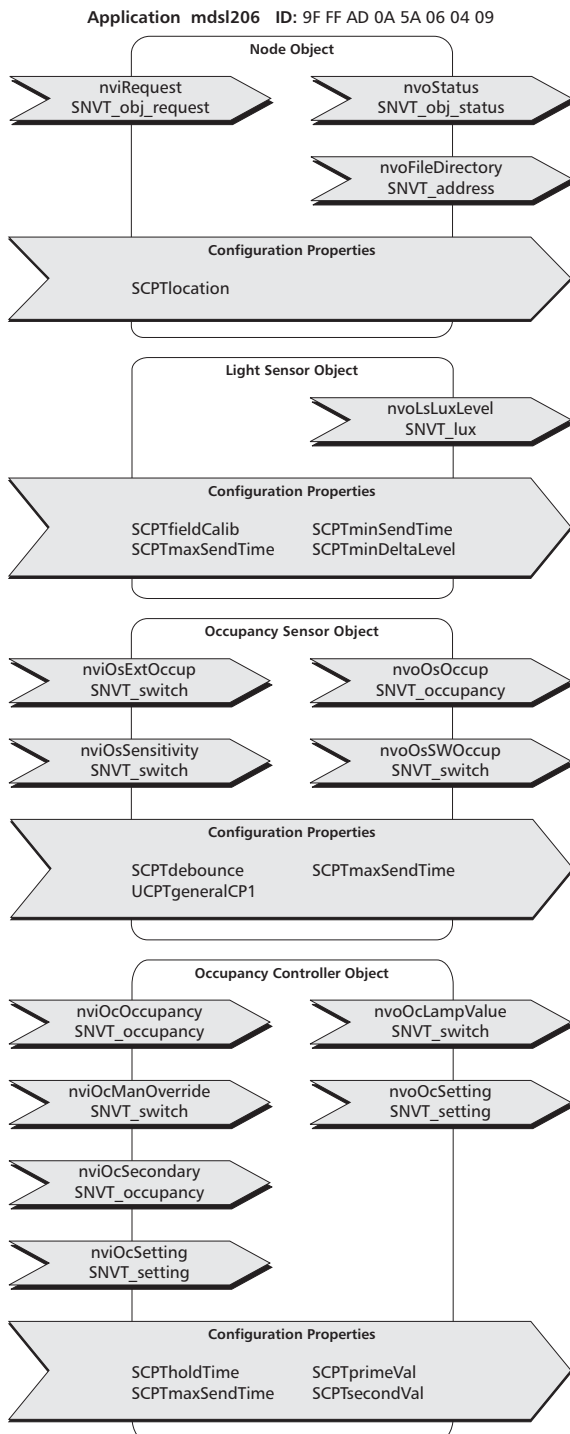


## Softwareapplikation mdsI206 (Sensorik, Constant Light Controller, Space Comfort Controller) Für Fühler Modell MDS-LON2 neue Ausführung

Der Decken Multisensor Modell MDS dient zur Bewegungserfassung, Helligkeitsmessung (0 - 1000 Lux) und Temperaturerfassung in Wohn- oder Büroräumen. Zusätzlich verfügt er über einen potentialfreien digitalen Eingang z.B. zum Anschluss eines Beleuchtungstasters oder Fensterkontaktes.

Die Vorgaben der LonMark® Funktionsprofile **1010** „Light Sensor“, **1060** „Occupancy Sensor“, **3071** „Occupancy Controller“, **3050** „Constant Light Controller“, **8500** „Space Comfort Controller“ und **3200** „Switch“ wurden berücksichtigt. Die Applikation verfügt über zwei identische Objekte 3050, so dass sich zwei Beleuchtungsgruppen unabhängig voneinander in einem Raum steuern lassen.

Die Applikation verwendet Standard-Netzwerkvariablen (SNVT) und Standard-Konfigurationsparameter (SCPT). 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 und sollten auf dem PC installiert werden, bevor das Installationstool die Gerätevorlagen erstellt.



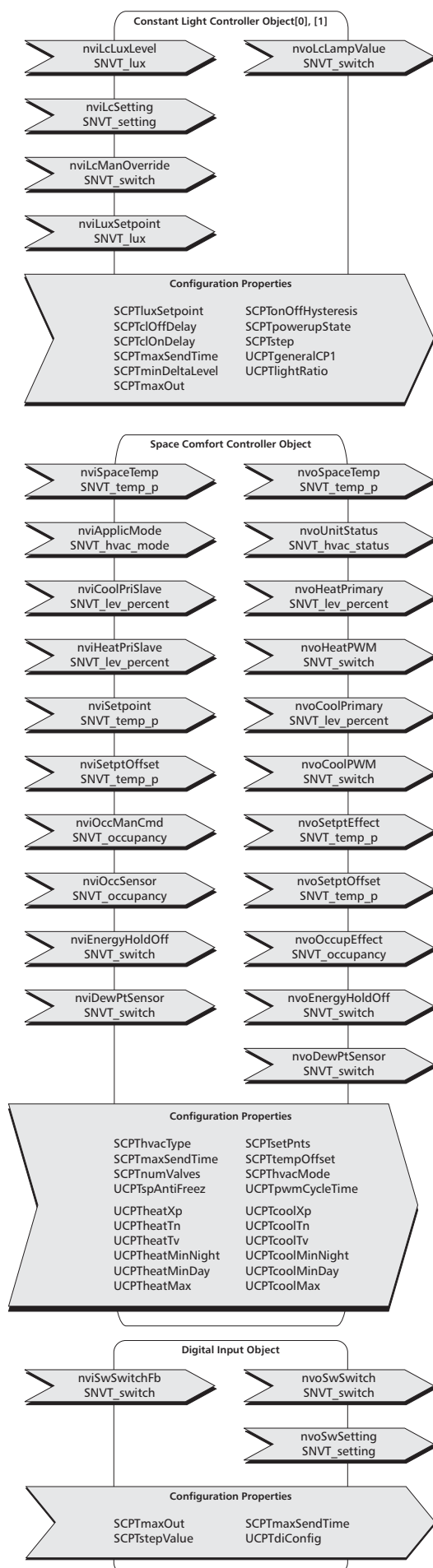
**Light Sensor:** Der gemessene Lichtwert wird mit der Variablen **nvoLsLuxLevel** ausgegeben. Zur Kalibrierung des Lichtsensors kann mit einem externen Luxmeter die exakte Beleuchtungsstärke ermittelt und über den Parameter **SCPTfieldCalib** eingegeben werden. Der Reflektionsfaktor wird dann automatisch berechnet und sowohl Messwert als auch Messbereichsendwert werden entsprechend korrigiert.

**Occupancy Sensor:** Die aktuelle Raumbelugung wird mit Variablen vom Typ SNVT\_occupancy und SNVT\_switch ausgegeben. Das Zurücksetzen der Ausgangsvariablen nach erkannter Bewegung erfolgt Zeitverzögert (einstellbar über **SCPTdebounce**).

Mit den Ein- / Ausgangsvariablen vom Typ SNVT\_switch bietet der Bewegungsmelder zusätzlich die Möglichkeit mehrere Bewegungsmelder miteinander zu verknüpfen oder direkt eine Beleuchtung bewegungsabhängig zu schalten. Mit **nviOsSensitivity** kann die Empfindlichkeit des Bewegungsmelders herabgesetzt werden. Die integrierte Bewegungsmelder-LED leuchtet immer für die Zeit der Initialisierungsphase. Ob die LED bei erkannter Bewegung leuchten soll ist über **UCPTgeneralCP1** einstellbar.

**Occupancy Controller:** Der Occupancy Controller kann als bewegungsabhängiger Beleuchtungschalter (mit **nvoOcLampValue**) oder zum Ein- / Ausschalten eines angeschlossenen Constant Light Controllers (mit **nvoOcSetting**) verwendet werden. Das Zurücksetzen der Ausgangsvariablen nach erkannter Bewegung erfolgt Zeitverzögert (einstellbar über **SCPTholdTime**). Die Eingangsvariable **nviOcOccupancy** kann mit der Ausgangsvariablen **nvoOsOccup** des **Occupancy Sensor Objects** (interner Bewegungsmelder) verbunden werden. Mit **nviOcOccupancy = OCCUPIED** wird die Beleuchtung auf den Wert **SCPTprimeVal** eingeschaltet. Die Eingangsvariable **nviOcSecondary** kann mit einem benachbarten Bewegungsmelder verbunden werden. Mit **nviOcSecondary = OCCUPIED** wird die Beleuchtung auf den Wert **SCPTsecondVal** eingeschaltet.

Über **nviOcSetting** kann der Controller aktiviert bzw. deaktiviert werden und mit **nviOcManOverride** besteht die Möglichkeit den Controller von extern zu übersteuern.



### Constant Light Controller Objekte [0] und [1]:

Zwei identische Objekte zur Beleuchtungssteuerung bzw. Beleuchtungsregelung. Mit **UCPTgeneralCP1** erfolgt die Konfigurierung der Controller. Sie können sowohl als Konstantlichtregler als auch als helligkeitsabhängige Bewegungsmelder verwendet werden. Konfiguriert als helligkeitsabhängige Bewegungsmelder können über die beiden Objekte zwei Beleuchtungsgruppen in einem Raum unabhängig voneinander kontrolliert und geschaltet werden. Die Eingangsvariable **nviLcLuxLevel** muss mit dem Lichtsensorobjekt und die Ausgangsvariable **nvoLcLampValue** mit einem Aktor zur Beleuchtungsansteuerung verbunden werden.

Mit **nviLcSetting** kann der Controller ein- bzw. ausgeschaltet und der Sollwert temporär verändert werden. Die Eingangsvariable **nviLcManOverride** dient zur manuellen Übersteuerung des Beleuchtungswertes.

### Space Comfort Controller Object:

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 **SCPTtempOffset** zur Verfügung.

Der effektive Sollwert (Basissollwert) **nvoSetptEffect** errechnet sich in Abhängigkeit der Eingangsvariablen zur Raumebelegung (**nviOccManCmd** und **nviOccSensor**), aus den Sollwertvorgaben über **SCPTsetPnts** bzw. **nviSetpoint** und dem Offsetwert **nviSetptOffset** (siehe Tabelle 1 auf Seite 13). Zur Auswertung des internen Bewegungsmelders muss die Ausgangsvariable **nvoOsOccup** des „Occupancy Sensor Objects“ mit der Eingangsvariablen **nviOccSensor** verbunden werden.

Für die Temperaturregelung verwendet der Regelalgorithmus den oben beschriebenen Basissollwert. Die neutrale (energiefreie) Zone um den Basissollwert passt sich automatisch der aktuellen Raumebelegung an und ist ebenfalls über **SCPTsetPnts** parametrierbar.

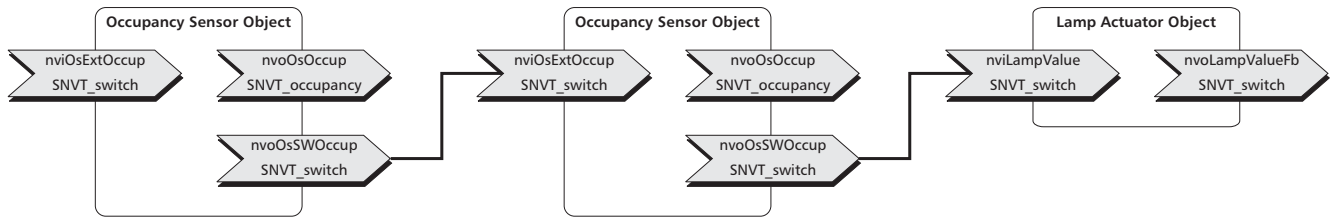
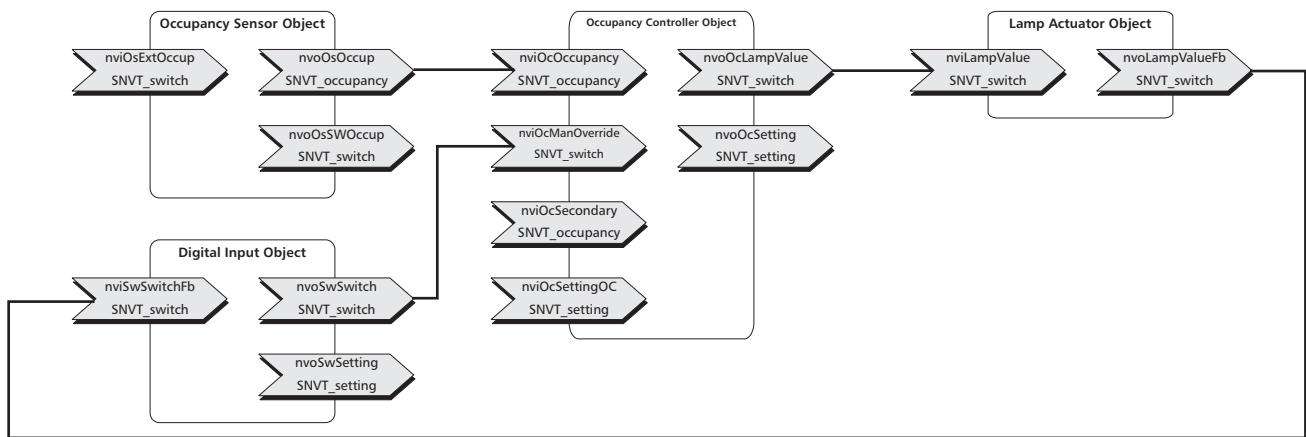
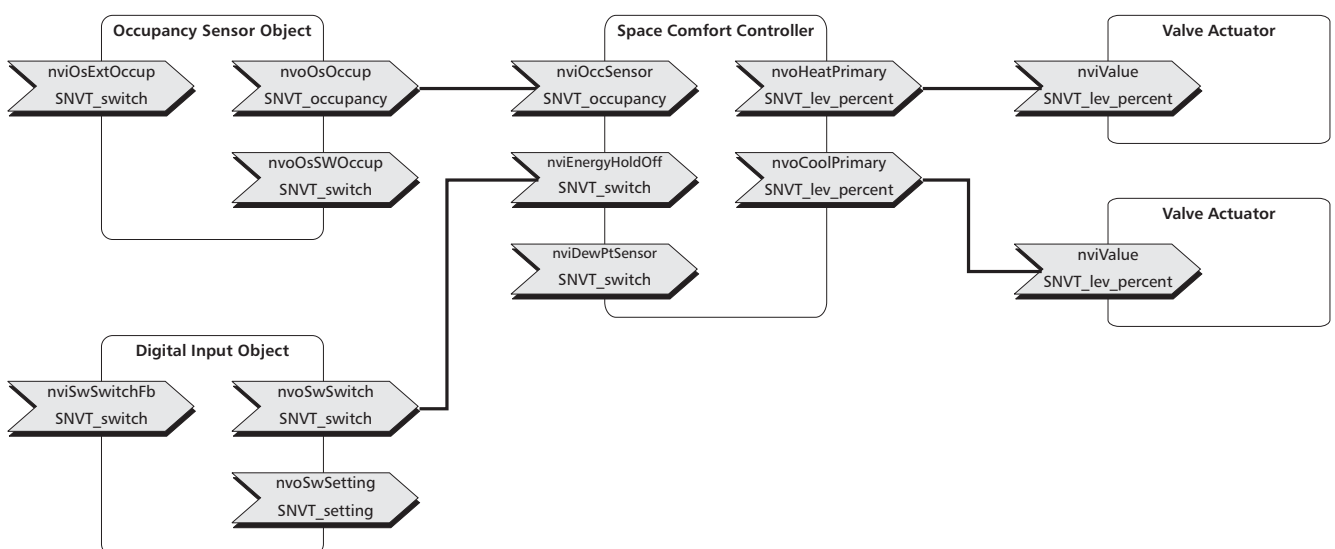
Die Stellgrößen des PID-Reglers für Heizen und Kühlen werden mit Variablen vom Typ **SNVT\_lev\_percent** für stetige Stellantriebe und mit Variablen vom Typ **SNVT\_switch** für thermische Zweipunkt-Stellantriebe (PWM-Ansteuerung) ausgegeben. Die Regelparameter Proportionalbereich, Nachstellzeit und Vorhaltezeit können hierbei den räumlichen Gegebenheiten individuell angepasst werden.

Die Überwachung von Fensterkontakt und Taupunktwärter erfolgt mit den Eingangsvariablen **nviEnergyHoldOff** und **nviDewPtSensor**.

### Digital Input Object:

Der Schaltzustand des potentialfreien digitalen Eingangs wird erfasst und je nach Konfiguration (UCPTdiConfig) über die Variablen vom Typ **SNVT\_switch** und **SNVT\_setting** ausgegeben. Die **SNVT\_switch**-Variable kann dem Beleuchtungsregler einen absoluten Beleuchtungswert zur manuellen Übersteuerung zur Verfügung stellen oder dem Temperaturregler den Status eines Fensterkontaktes bzw. Taupunktwärters übermitteln. Mit **SNVT\_setting** kann der Occupancy Controller oder der Constant Light Controller aktiviert bzw. deaktiviert werden.

Die digitalen Eingänge können die Funktionen Standard I/O, Toggle, Dimmen oder Automatik übernehmen. Mit **SCPTmaxOut** lässt sich der maximale Ausgabewert der **SNVT\_switch** Variablen begrenzen.

**Anwendungsbeispiele:****Occupancy Sensor:****ODER-Verknüpfung von 2 Bewegungsmeldungen und direkte Beleuchtungsansteuerung****Occupancy Controller:****Bewegungsabhängiges Schalten der Beleuchtung mit manueller Übersteuerung über Taster mit Toggle-Funktion****Space Comfort Controller:****Sollwertauswahl in Abhängigkeit der Raumbellegung und Auswertung eines Fensterkontakts am digitalen Eingang**

**Constant Light Controller:****Hinweise zur Kalibrierung und Inbetriebnahme:**

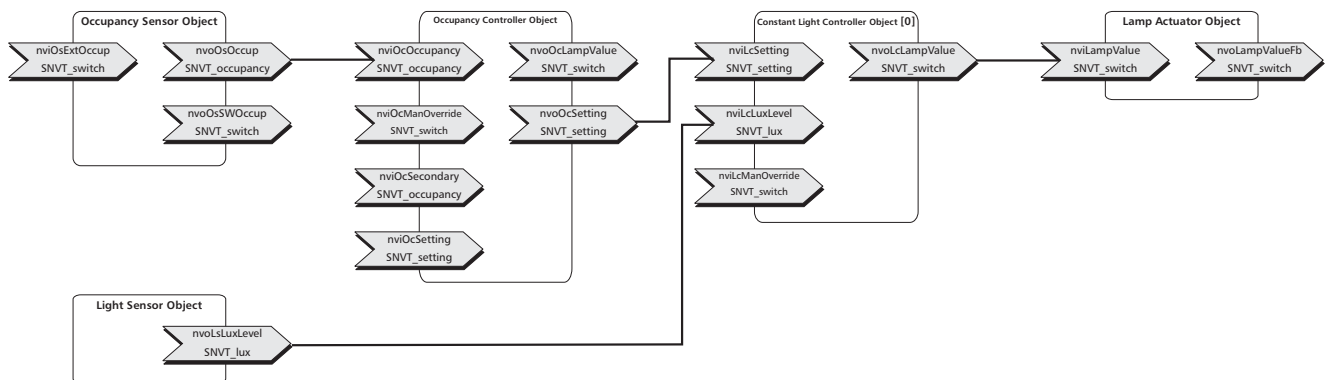
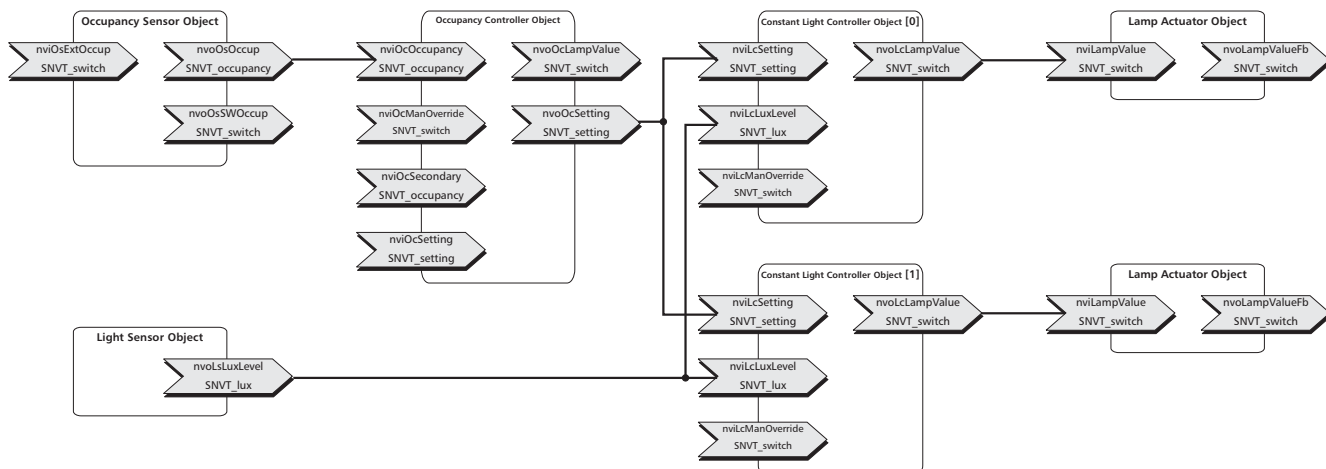
**1. Light Sensor:** Die Kalibrierung des Lichtfühlers sollte ohne Kunstlicht und bei einer raumtypischen Jalousieposition durchgeführt werden. Die Beleuchtungsstärke wird hierzu auf der Arbeitsfläche mit einem Referenzgerät gemessen und dann als Konfigurationswert in den Parameter **SCPTfieldCalib** eingetragen.

**2. Constant Light Controller:** Da die Empfindlichkeit des Lichtfühlers gegenüber Kunstlicht u.a. von der verwendeten Lichtquelle, der Platzierung des Sensors und den Reflektionseigenschaften des Raumes abhängt, muss zur Inbetriebnahme der Lichtregelung diese Empfindlichkeit mit dem Konfigurationsparameter **UCPTlightRatio** bestimmt werden.

**UCPTlightRatio.multiplier:** Beleuchtungsstärke bei 100 % Kunstlicht, gemessen auf der Arbeitsfläche mit einem Referenzgerät.

**UCPTlightRatio.divisor:** Beleuchtungsstärke bei 100 % Kunstlicht, gemessen mit dem Multisensor MDS LON1.

Die Bestimmung der Werte muss bei gut abgedunkeltem Raum, möglichst ohne Tageslichteinfluss erfolgen. Das Kunstlicht sollte dabei mit 100 % auf seinen Maximalwert geschaltet sein.

**Konstantlichtregelung einer Beleuchtungsgruppe mit Aktivierung über Occupancy Controller****Helligkeitsabhängige Bewegungsmelder zum Schalten von zwei Beleuchtungsgruppen mit Aktivierung über Occupancy Controller**

## Node Object

Das Node Objekt überwacht und steuert die Funktionen der einzelnen Objekte im Gerät. Unterstützt wird die von LonMark® geforderte Grundfunktionalität.

### Variablen Node Object:

#### nviRequest

SNVT Typ: SNVT\_obj\_request, Index 92

Funktion: Eingangsvariable mit den Funktionen RQ\_NORMAL, RQ\_UPDATE\_STATUS und RQ\_REPORT\_MASK.

#### 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.

## Light Sensor Object

Das Objekt beinhaltet die Funktionen zur Messung der Beleuchtungsstärke und Datenausgabe.

### Ausgangsvariablen Light Sensor Object:

#### nvoLsLuxLevel

SNVT Typ: SNVT\_lux, Index 79

Funktion: Ausgangsvariable für die gemessene Beleuchtungsstärke in Lux. Die Datenausgabe erfolgt in Abhängigkeit der Konfigurationsparameter *SCPTminSendTime*, *SCPTmaxSendTime* und *SCPTminDeltaLevel* und 1,5s- 4s nach Reset.

### Konfigurationsparameter Light Sensor Object:

#### SCPTfieldCalib

SCPT Index: 90, SNVT\_lux

Funktion: Konfigurationsparameter zur Selbstkalibrierung des Lichtsensors. Mit einem externen Luxmeter kann die exakte Beleuchtungsstärke ermittelt und eingegeben werden. Der Reflektionsfaktor wird dann automatisch berechnet und sowohl Messwert als auch Messbereichsendwert werden entsprechend korrigiert.

**!! Die Kalibrierung sollte ohne Kunstlicht und bei einer raumtypischen Jalousieposition !! durchgeführt werden. Der Kalibrierwert darf nur eingegeben werden, wenn der Sensor direkt am !! Netzwerk angeschlossen ist, da ansonsten der Korrekturwert nicht berechnet wird.**

(Voreingestellter Wert: 0 Lux ==> Field Calibration deaktiviert)

#### SCPTmaxSendTime

SCPT Index: 49, SNVT\_time\_sec

Funktion: Heartbeatfunktion. Legt die Intervallzeit fest, nach der die Ausgangsvariablen unabhängig einer Wertänderung gesendet werden. Mit Eingabewert = 0 wird die Heartbeatfunktion deaktiviert. (Voreingestellter Wert: 60 sec.)

#### SCPTminSendTime

SCPT Index: 52, SNVT\_time\_sec

Funktion: Legt das kleinste Update-Intervall der Ausgangsvariable fest. Ein Update erfolgt nach Ablauf von *SCPTminSendTime*, wenn sich der Lichtwert um mindestens *SCPTminDeltaLevel* geändert hat. Mit Eingabewerten = 0 wird die „Minsend“-Funktion deaktiviert. (Voreingestellter Wert: 1 sec.)



***SCPTminDeltaLevel***

SCPT Index: 88, SNVT\_lev\_cont

Funktion: Wenn sich die Beleuchtungsstärke um den eingestellten Wert ***SCPTminDeltaLevel*** verändert (% vom Messbereich), dann werden die neuen Lichtwerte übertragen. Die Funktion ist abhängig von der Einstellung ***SCPTminSendTime***. (Wertebereich: 0 % - 100 %; Voreingestellter Wert: 2,5 %)

***Occupancy Sensor Object******Eingangsvariablen Occupancy Sensor Object:******nviOsExtOccup***

SNVT Typ: SNVT\_switch, Index 95

Funktion: Eingangsvariable für externe Bewegungsmelder (z.B. ODER-Verknüpfung mehrerer Bewegungsmelder). Bei ***nviOsExtOccup*** = 100,0 1 werden die Ausgangsvariablen auf OC\_OCCUPIED bzw. 100,0 1 gesetzt. Bei anderen Werten werden die Ausgangsvariablen nach Ablauf der Verzögerungszeit ***SCPTdebounce*** zurückgesetzt. Der interne IR-Bewegungsmelder ist mit der Ansteuerung über ***nviOsExtOccup*** ODER-Verknüpft.

***nviOsSensitivity***

SNVT Typ: SNVT\_switch, Index 95

Funktion: Mit ***nviOsSensitivity*** = 0,0 0 kann die Empfindlichkeit des Bewegungsmelders herabgesetzt werden. Initialisierungswert nach Reset: ***nviOsSensitivity*** = 100,0 1, d.h. hohe Empfindlichkeit

***Ausgangsvariablen Occupancy Sensor Object:******nvoOsOccup***

SNVT Typ: SNVT\_occupancy, Index 109

Funktion: Ausgangsvariable Bewegungsmeldung. Wird gesetzt sobald eine interne oder externe Bewegung erkannt wurde. Das Rücksetzen erfolgt nach Ablauf der Verzögerungszeit ***SCPTdebounce***. Die Datenübertragung erfolgt in Abhängigkeit der Konfigurationsparameter ***SCPTdebounce*** und ***SCPTmaxSendTime***.

**Modul-Reset:** Für die ersten 60 sec. nach Reset (Initialisierungsphase des Bewegungsmelders) erfolgt keine Datenübertragung und ***nvoOccup*** erhält den Wert OC\_UNOCCUPIED.

***nvoOsSWOccup***

SNVT Typ: SNVT\_switch, Index 95

Funktion: Ausgangsvariable Bewegungsmeldung. Wird parallel mit ***nvoOsOccup*** gesendet. Diese Variable kann als „externe Bewegungsmeldung“ von einem weiteren Bewegungsmelder ausgewertet werden, oder direkt eine Beleuchtungsgruppe ansteuern.

***Konfigurationsparameter Occupancy Sensor Object:******SCPTmaxSendTime***

SCPT Index: 49, SNVT\_time\_sec

Funktion: Heartbeatfunktion. Legt die Intervallzeit fest, nach der die Ausgangsvariablen unabhängig einer Ergebnisänderung gesendet werden. Mit Eingabewert = 0 wird die Heartbeatfunktion deaktiviert. (Voreingestellter Wert: 120 sec)

***SCPTdebounce***

SCPT Index: 139, SNVT\_time\_sec

Funktion: Zeitverzögerung für das Zurücksetzen der Ausgangsvariablen nach erkannter Bewegung. Der Verzögerungstimer wird nach Zustandswechsel „Bewegung ==> Keine Bewegung“ gestartet. (Voreingestellter Wert: 0 sec.)

***UCPTgeneralCP1***

UCPT Index: 7, SNVT\_state

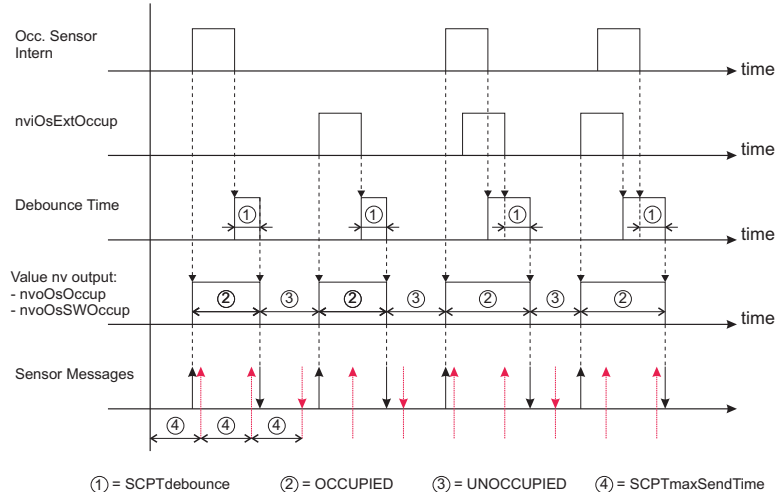
Funktion: Die integrierte Bewegungsmelder-LED leuchtet immer für die Zeit der Initialisierungsphase. Ob die LED bei erkannter Bewegung leuchten soll ist über ***UCPTgeneralCP1*** einstellbar.

UCPTgeneralCP1.bit0 = 0 ==> LED leuchtet nur in der Initialisierungsphase des Gerätes

UCPTgeneralCP1.bit0 = 1 ==> LED blinkt zusätzlich bei erkannter Bewegung

**UCPTgeneralCP1 (Fortsetzung)**

Funktion: Parameter für die Ausgangsvariable `nvoOcSetting` im Occupancy-Controller-Objekt.  
 UCPTgeneralCP1.bit15 = 0 ==> `nvoOcSetting.function` sendet SET\_ON und SET\_OFF (default)  
 UCPTgeneralCP1.bit15 = 1 ==> `nvoOcSetting.function` sendet nur SET\_OFF

**Funktionsdiagramm Occupancy Sensor:****Occupancy Controller Object**

Der Occupancy Controller kann mit ***nvoOcLampValue*** als bewegungsabhängiger Beleuchtungsschalter oder mit ***nvoOcSetting*** zum Ein- / Ausschalten eines angeschlossenen Constant Light Controllers verwendet werden.

**Eingangsvariablen Occupancy Controller Object:*****nviOcOccupancy***

SNVT Typ: SNVT\_occupancy, Index 109

Funktion: Die Eingangsvariable ***nviOcOccupancy*** stellt dem Controller die aktuelle Raumbesetzung zur Verfügung und wird mit der Ausgangsvariable ***nvoOsOccup*** des Occupancy Sensors verbunden.  
 (Initialisierungswert nach Reset: OC\_NUL)

***nviOcManOverride***

SNVT Typ: SNVT\_switch, Index 95

Funktion: Eingangsvariable zur manuellen Steuerung der Beleuchtung, unabhängig des Variablenwertes ***nviOcSetting***. Ein Update von ***nviOcManOverride*** sperrt den Controller und die Ausgangsvariable ***nvoOcLampValue*** übernimmt die Werte von ***nviOcManOverride***.

`nviOcManOverride.state = 0` ==> `nvoOcLampValue = 0.0 0`

`nviOcManOverride.state = 1` ==> `nvoOcLampValue = nviOcManOverride`

Sollte der Occupancy Controller durch ***nviOcManOverride*** deaktiviert sein, dann wird der Controller nach Empfang von UNOCCUPIED an ***nviOcOccupancy*** und Ablauf der Verzögerungszeit ***SCPTholdTime*** wieder in den Automatikmodus zurückgeschaltet. (Initialisierungswert nach Reset: 0.0 -1)

***nviOcSecondary***

SNVT Typ: SNVT\_occupancy, Index 109

Funktion: Eingangsvariable eines benachbarten Bewegungsmelders mit der aktuellen Raumbesetzung eines benachbarten Gebietes. (Initialisierungswert nach Reset: OC\_NUL)

***nviOcSetting***

SNVT Typ: SNVT\_setting, Index 117

Funktion: Die Eingangsvariable ***nviOcSetting*** aktiviert bzw. deaktiviert den Controller.

Initialisierungszustand nach Reset: `nviSettingOC.function = SET_ON`

`nviOcSetting.function = SET_OFF` ==> Controller = AUS; `nvoOcLampValue = 0.0 0` (Beleuchtung AUS)

`nviOcSetting.function = SET_ON` ==> Controller = EIN;

**Ausgangsvariablen Occupancy Controller Object:****nvoOcLampValue**

SNVT Typ: SNVT\_switch, Index 95

Funktion: Ausgangsvariable zur Ansteuerung der Beleuchtung. (siehe Funktionsdiagramm Occupancy Controller).

nvoOcLampValue.state = 0 ==&gt; Beleuchtung AUS

nvoOcLampValue.state = 1 ==&gt; Beleuchtung EIN

nvoOcLampValue.value = Beleuchtungsstärke (0 - 100 %)

Die Datenausgabe erfolgt in Abhängigkeit des Konfigurationsparameters **SCPTmaxSendTime**, bei Änderung des Ausgabewertes und 1,5s- 4s nach Reset.**nvoOcSetting**

SNVT Typ: SNVT\_setting, Index 117

Funktion: Ausgangsvariable zur Steuerung eines nachgeschalteten Controllers, z.B. Constant Light Controller (siehe Funktionsdiagramm Occupancy Controller). Die Datenausgabe erfolgt analog zu **nvoOcLampValue**.

nviOccupancy oder nviSecondary = OCCUPIED ==&gt; nvoSettingOC.function = SET\_ON

nviOccupancy und nviSecondary = UNOCCUPIED ==&gt; nvoSettingOC.function = SET\_OFF

**Konfigurationsparameter Occupancy Controller Object:****SCPTholdTime**

SCPT Index: 91, SNVT\_time\_sec

Funktion: Zeitverzögerung für das Zurücksetzen der Ausgangsvariablen **nvoOcLampValue** und **nvoOcSetting** nachdem **nviOcOccupancy** und **nviOcSecondary** den Zustand UNOCCUPIED eingenommen haben. Der Verzögerungstimer wird nach Zustandswechsel „OCCUPIED ==> UNOCCUPIED“ gestartet.  
(Voreingestellter Wert: 600,0 sec = 10 min)**SCPTprimeVal**

SCPT Index: 155, SNVT\_switch

Funktion: Der Konfigurationsparameter **SCPTprimeVal** definiert den Ausgabewert von **nvoLampValueOC** wenn nviOcOccupancy = OC\_OCCUPIED. (Voreingestellter Wert: 100.0 1)**SCPTsecondVal**

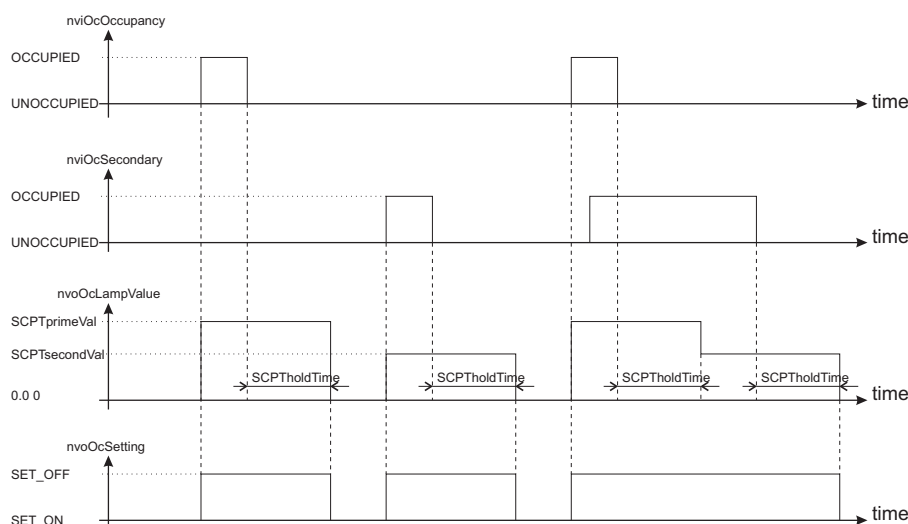
SCPT Index: 156, SNVT\_switch

Funktion: Der Konfigurationsparameter **SCPTsecondVal** definiert den Ausgabewert von **nvoLampValueOC** wenn nviOccupancy = UNOCCUPIED und nviSecondary = OCCUPIED. (Voreingestellter Wert: 0.0 0)**SCPTmaxSendTime**

SCPT Index: 49, SNVT\_time\_sec

Funktion: Heartbeatfunktion. Legt die Intervallzeit fest, nach der die Ausgangsvariablen unabhängig einer Ergebnisänderung gesendet werden.

Mit Eingabewert = 0 wird die Heartbeatfunktion deaktiviert. (Voreingestellter Wert: 120 sec)

**Funktionsdiagramm Occupancy Controller:**



## Constant Light Controller Object [0], [1]

Zwei identische Objekte zur Regelung der Beleuchtungsstärke auf einen vorgegebenen Sollwert. Über **UCPTgeneralCP1** kann die Funktion des Constant Light Controllers in die eines helligkeitsabhängigen Bewegungsmelders zur Beleuchtungssteuerung (Ein- bzw. Ausschalten der Beleuchtung in Abhängigkeit der Raumbelastung und der Helligkeit) umgeschaltet werden.

**!! Konfiguriert als helligkeitsabhängige Bewegungsmelder können die beiden Lichtregler-Objekte zusammen mit dem Lichtsensor des MDS zwei Beleuchtungsgruppen in einem Raum unabhängig voneinander kontrollieren und schalten.**

**!! Konfiguriert als Konstantlichtregler benötigt das zweite Lichtregler-Objekt zur Regelung einen zusätzlichen Lichtsensor.**

### Eingangsvariablen Constant Light Controller Object:

#### nviLcLuxLevel[0], [1]

SNVT Typ: SNVT\_lux, Index 79

Funktion: Die Eingangsvariable enthält die aktuelle Beleuchtungsstärke im Raum und wird mit der Ausgangsvariablen **nvoLcLuxLevel** des Light Sensors verbunden.

#### nviLcSetting[0], [1]

SNVT Typ: SNVT\_setting, Index 117

Funktion: Die Eingangsvariable bestimmt den Betriebsstatus des Reglers (EIN oder AUS) und kann zusätzlich zur temporären Sollwertverstellung verwendet werden.

nviLcSetting.function = SET\_ON: Regler = EIN, d.h. die Ausgangsgröße zur Beleuchtungssteuerung (nvoLcLampValue) wird so verändert, dass die Beleuchtungsstärke im Raum dem eingestellten Sollwert entspricht.

nviLcSetting.function = SET\_OFF: Regler = AUS und Beleuchtung AUS (nvoLcLampValue = 0.0 0)

Bei Konfiguration als Konstantlichtregler (**UCPTgeneralCP1.bit14 = 0** und **UCPTgeneralCP1.bit15 = 0**):

nviLcSetting.function = SET\_UP: Erhöhen der Ausgangsvariablen nvoLcLampValue.value um den Wert nviLcSetting.setting. Der neue Lichtwert wird automatisch neuer Beleuchtungssollwert.

nviLcSetting.function = SET\_DOWN: Verkleinern der Ausgangsvariablen nvoLcLampValue.value um den Wert nviLcSetting.setting. Der neue Lichtwert wird automatisch neuer Beleuchtungssollwert.

Mit einem Update auf SET\_ON wird der Sollwert wieder auf den Basissollwert SCPTluxSetpoint zurückgesetzt.

#### nviLcManOverride[0], [1]

SNVT Typ: SNVT\_switch, Index 95

Funktion: Eingangsvariable zur manuellen Steuerung der Beleuchtung. Initialisierungswert nach Reset: 0.0 -1

Bei Konfiguration als Konstantlichtregler (**UCPTgeneralCP1.bit14 = 0**, **UCPTgeneralCP1.bit15 = 0**) oder als helligkeitsabhängiger Bewegungsmelder zur Beleuchtungssteuerung (**UCPTgeneralCP1.bit14 = 0**, **UCPTgeneralCP1.bit15 = 1**):

Ein Update von nviLcManOverride sperrt den Controller und die Ausgangsvariable nvoLcLampValue übernimmt die Werte von nviLcManOverride. Mit nviLcManOverride.state = -1 wird der Controller wieder aktiviert.

nviLcManOverride.state = -1 ==> Lichtregler EIN

nviLcManOverride.state = 0, 1 und .value = 0 - 100 % ==> Lichtregler AUS

==> nvoLcLampValue = nviLcManOverride

Bei Konfiguration zum helligkeitsabhängigen Ausschalten der Beleuchtung (**UCPTgeneralCP1.bit14 = 1** und **UCPTgeneralCP1.bit15 = 0**):

Mit einem Update von nviLcManOverride = 100.0 1 wird die Beleuchtung mit nvoLcLampValue = 100.0 1 eingeschaltet. Ausgeschaltet wird die Beleuchtung dann in Abhängigkeit der gemessenen Beleuchtungsstärke im Raum (siehe Funktionsdiagramm auf Seite 10).

**Automatisches Zurücksetzen auf Automatikbetrieb (0.0 -1) wenn UCPTgeneralCP1.bit1 = 1**

Bei einem Wechsel des Variablenwertes **nviLcSetting.function** auf **SET\_OFF** wird die Variable nviLcManOverride unabhängig von der Reglerkonfiguration zurückgesetzt auf 0.0 -1.

***nviLuxSetpoint[0], [1]***

SNVT Typ: SNVT\_lux, Index 79

Funktion: Eingangsvariable zur Vorgabe des Sollwertes zur Beleuchtungsregelung. Nach Reset übernimmt nviLuxSetpoint die Einstellungen aus **SCPTluxSetpoint**.***Ausgangsvariablen Constant Light Controller Object:******nvoLcLampValue[0], [1]***

SNVT Typ: SNVT\_switch, Index 95

Funktion: Ausgangsvariable zur Ansteuerung der Beleuchtung.

nvoLcLampValue.state = 0 ==&gt; Beleuchtung AUS

nvoLcLampValue.state = 1 ==&gt; Beleuchtung EIN

nvoLcLampValue.value = Beleuchtungsstärke (0 - 100 %)

Die Datenausgabe erfolgt in Abhängigkeit der Konfigurationsparameter **SCPTminDeltaLevel**, **SCPTmaxSendTime** und 1,5s- 4s nach Reset.***Konfigurationsparameter Constant Light Controller Object:******SCPTluxSetpoint***

SCPT Index: 82, SNVT\_lux

Funktion: Initialisierungswert von nviLuxSetpoint, zu Vorgabe des Sollwertes zur Beleuchtungsregelung. (Voreingestellter Wert: 500 lux)

***SCPTcIOffDelay***

SCPT Index: 85, SNVT\_time\_sec

Funktion: Ausschaltverzögerung für die Beleuchtung (nvoLcLampValue.state = 0).

Überschreitet der Lichtwert die Grenze (SCPTluxSetpoint + SCPTonOffHysteresis/2) für die Zeit SCPTcIOffDelay, dann wird die Beleuchtung ausgeschaltet. Bei Konfiguration als Konstantlichtregler wird mit **SCPTonOffHysteresis** = 0 das autom. Ausschalten deaktiviert. (Voreingestellter Wert: 300,0 sec = 5 min)***SCPTcIOnDelay***

SCPT Index: 86, SNVT\_time\_sec

Funktion: Einschaltverzögerung für die Beleuchtung (nvoLcLampValue.state = 1).

Wird der Grenzwert (SCPTluxSetpoint - SCPTonOffHysteresis/2) für die Zeit SCPTcIOnDelay unterschritten, dann wird die Beleuchtung eingeschaltet. (Voreingestellter Wert: 0 sec)

***SCPTmaxSendTime***

SCPT Index: 49, SNVT\_time\_sec

Funktion: Heartbeatfunktion. Legt die Intervallzeit fest, nach der die Ausgangsvariablen unabhängig einer Ergebnisänderung gesendet werden.

Mit Eingabewert = 0 wird die Heartbeatfunktion deaktiviert. (Voreingestellter Wert: 300 sec = 5 min)

***SCPTminDeltaLevel***

SCPT Index: 88, SNVT\_lev\_cont

Funktion: Wenn sich der Ausgabewert um den eingestellten Wert SCPTminDeltaLevel verändert, dann wird nvoLcLampValue gesendet. (Wertebereich: 0 % - 100 %; Voreingestellter Wert: 0,5 %)

***SCPTonOffHysteresis***

SCPT Index: 84, SNVT\_lev\_cont

Funktion: Relativer Hysteresewert (% von SCPTluxSetpoint) zur Berechnung der Schaltschwellen an denen die Beleuchtung in Abhängigkeit der Verzögerungszeiten **SCPTcIOnDelay** und **SCPTcIOffDelay** ein- bzw. ausgeschaltet wird. Bei Konfiguration als Konstantlichtregler wird mit **SCPTonOffHysteresis** = 0 das autom. Ausschalten deaktiviert. (Voreingestellter Wert: 0 %).Automatisches einschalten:  $nviLcLuxLevel < SCPTluxSetpoint - SCPTonOffHysteresis/2$ Automatisches ausschalten:  $nvoLcLampValue.value = 0$   
und $nviLcLuxLevel > SCPTluxSetpoint + SCPTonOffHysteresis/2$ ***SCPTpowerupState***

SCPT Index: 87, SNVT\_setting

Funktion: Initialisierungswert für den Betriebsmodus des Reglers nach Reset. (Voreingestellter Wert: {SET\_OFF,0,0})

**UCPTlightRatio**

UCPT Index: 11, SNVT\_muldiv

Funktion: Konfigurationsparameter zur Bestimmung der Empfindlichkeit des Lichtfühlers gegenüber Kunstlicht.  
Voreingestellter Wert: 1000, 100

UCPTlightRatio.multiplier: Beleuchtungsstärke bei 100 % Kunstlicht, gemessen auf der Arbeitsfläche mit einem Referenzgerät.

UCPTlightRatio.divisor: Beleuchtungsstärke bei 100 % Kunstlicht, gemessen mit dem Multisensor MDS über *nvoLsLuxLevel*.

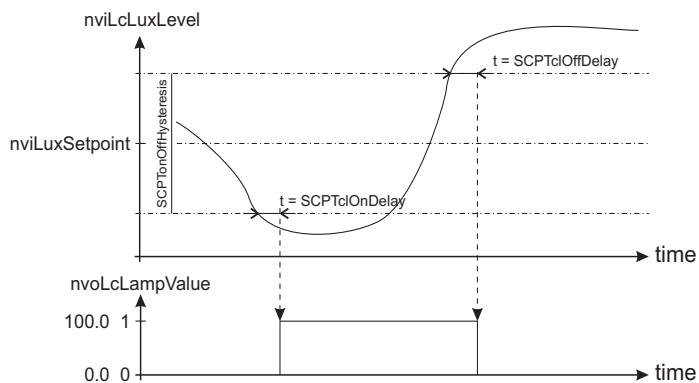
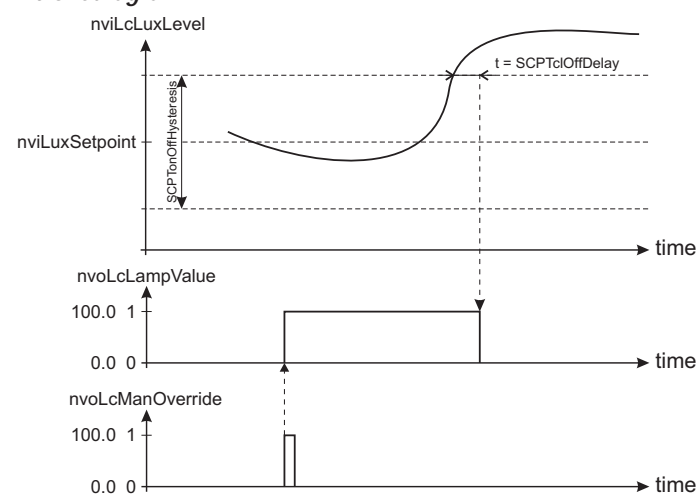
**!! Die Bestimmung der Werte muss bei gut abgedunkeltem Raum, möglichst ohne Tageslichteinfluss !! erfolgen. Das Kunstlicht sollte mit 100 % auf seinen Maximalwert geschaltet sein.**

**UCPTgeneralCP1**

UCPT Index: 7, SNVT\_state

Funktion: Konfigurationsparameter zur Einstellung der Reglerfunktionalität.

(Voreingestellter Wert: UCPTgeneralCP1 = 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0 d.h. Konstantlichtregelung)

UCPTgeneralCP1 .bit14	UCPTgeneralCP1 .bit15	Funktion
0	0	Konstantlichtregelung ( <b>Constant LC</b> ) UCPTgeneralCP1.bit0 = 0: Startwert nvoLcLampValue=SCPTmaxOut UCPTgeneralCP1.bit0 = 1: Startwert nvoLcLampValue=0%
0	1	Helligkeitsabhängiger Bewegungsmelders zur Beleuchtungssteuerung ( <b>ON/OFF LC</b> ) <b>Funktionsdiagramm:</b> 
1	0	Helligkeitsabhängiges Ausschalten der Beleuchtung ( <b>OFF LC</b> ). Das Einschalten der Beleuchtung erfolgt mit nviLcManOverride = 100 1 <b>Funktionsdiagramm:</b> 

**UCPTgeneralCP1 (Fortsetzung)**

Zusatzfunktion UCPTgeneralCP1.bit1 (Voreingestellter Wert = 0)

UCPTgeneralCP1.bit1 = 1: *nviLcManOverride* wird bei einem Wechsel von *nviLcSetting.function* auf SET\_OFF zurückgesetzt auf den Initialisierungswert 0.0 -1.

UCPTgeneralCP1.bit1 = 0: *nviLcManOverride* wird nicht automatisch verändert. Änderungen müssen über das Netzwerk vorgegeben werden.

**SCPTstep**

SCPT Index: 83, SNVT\_lev\_cont

Funktion: Maximale Schrittweite mit der die Ausgangsvariable *nvoLcLampValue.value* vom Lichtregler verändert werden darf um den eingestellten Sollwert zu erreichen.  
(Voreingestellter Wert: 10 %)

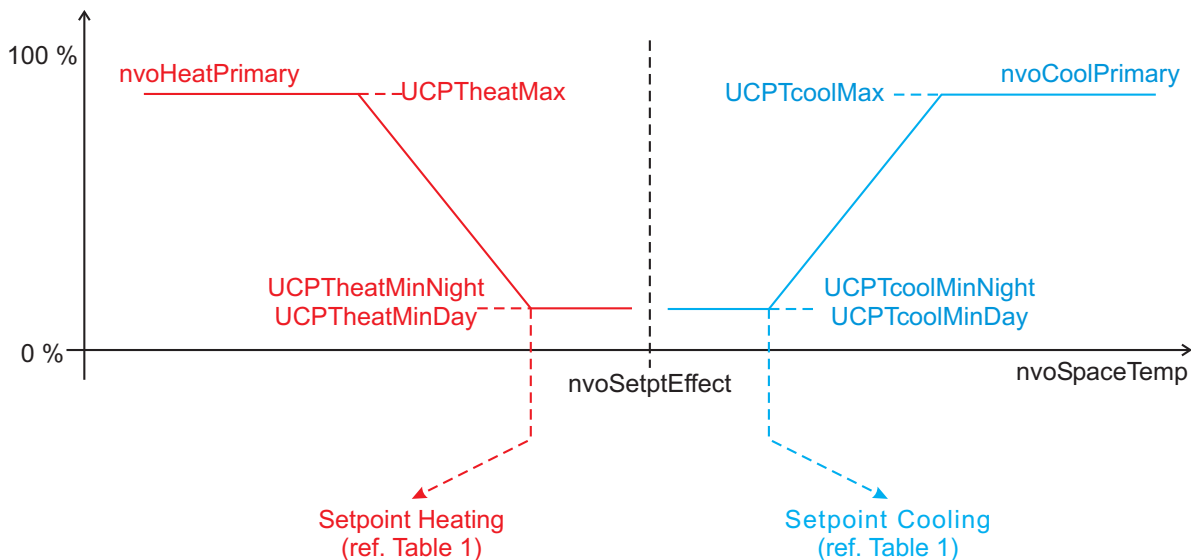
**SCPTmaxOut**

SCPT Index: 93, SNVT\_lev\_cont

Funktion: Konfigurationsparameter zur Begrenzung des Ausgabewertes der Ausgangsvariablen *nvoLcLampValue*, wenn der Regler den Lichtwert automatisch verändert. Bei einer externen Vorgabe über *nviLcManOverride* wird der Ausgabewert von *nvoLcLampValue* nicht begrenzt. (Voreingestellter Wert: 100 %)

**Space Comfort Controller Object**

Das Objekt beinhaltet die Funktionen Temperaturmessung und PID-Regelung für Heizen und Kühlen. Die Ausgabe der Stellgrößen erfolgt über Netzwerkvariablen. Die Abtastzeit zur Stellgrößenberechnung entspricht dem Sendeintervall (SCPTmaxSendTime) der Ausgangsvariablen.

**Funktionsdiagramm PID-Regler Heizen/Kühlen:****Eingangsvariablen Space Comfort Controller Object****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!

**nviApplicMode**

SNVT Typ: SNVT\_hvac\_mode, Index 108

Funktion: Eingangsvariable zur Auswahl des Betriebsmodus des Reglers.

- HVAC\_AUTO ==> automatisches Umschalten zwischen Heizen und Kühlen
- HVAC\_HEAT ==> nur Heizen
- HVAC\_COOL ==> nur Kühlen
- HVAC\_OFF ==> Regelung ausgeschaltet

Der Initialisierungszustand nach Reset wird durch den Konfigurationsparameter *SCPTvacMode* bestimmt.

**nviCoolPriSlave**

SNVT Typ: SNVT\_lev\_percent, Index 81

Funktion: Steuervariable für die Netzwerkvariablen **nvoCoolPrimary** und **nvoCoolPWM**.

nviCoolPriSlave = 0x7FFF (163,835 %) ==&gt; interner Regler Kühlen EIN (Initialisierungswert nach Reset)

nviCoolPriSlave = 0 ... 100 % ==&gt; interner Regler Kühlen AUS

==&gt; nviCoolPriSlave bestimmt die Ausgangsgrößen

**!! Die externe Übersteuerung hat höchste Priorität, auch eine gleichzeitige Ansteuerung von Heiz-  
!! und Kühlventil ist daher möglich.****nviHeatPriSlave**

SNVT Typ: SNVT\_lev\_percent, Index 81

Funktion: Steuervariable für die Netzwerkvariable **nvoHeatPrimary** und **nvoHeatPWM**.

nviHeatPriSlave = 0x7FFF (163,835 %) ==&gt; interner Regler Heizen EIN (Initialisierungswert nach Reset)

nviHeatPriSlave = 0 ... 100 % ==&gt; interner Regler Heizen AUS

==&gt; nviHeatPriSlave bestimmt die Ausgangsgrößen

**!! Die externe Übersteuerung hat höchste Priorität, auch eine gleichzeitige Ansteuerung von Heiz-  
!! und Kühlventil ist daher möglich.****nviOccManCmd und nviOccSensor**

SNVT Typ: SNVT\_occupancy, Index 109

Funktion: Eingangsvariablen zur Vorgabe der Raumbellegung. Die aktuelle Raumbellegung 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 mit: OC\_OCCUPIED, OC\_STANDBY, OC\_UNOCCUPIED

nviOccSensor: Präsenzmeldung im Raum mit: OC\_OCCUPIED, OC\_UNOCCUPIED

nviOccManCmd	nviOccSensor	>>>	room occupancy nvoOccupEffect	Setpoint Heat nvoSetptEffect (nvoUnitStatus.mode = HVAC_HEAT)	Setpoint 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

**Tabelle 1: Regelparameter in Abhängigkeit der Raumbellegung****nviSetpoint**

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 werden zur Berechnung des effektiven Sollwertes (Basis-Sollwert + Offset) die Werte des Konfigurationsparameters **SCPTsetPnts** 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 Vorgabe eines Offsetwertes zur Verschiebung der Basis-Sollwerttemperatur in den Modi OCCUPIED oder STANDBY (siehe Tabelle 1).

### ***nviEnergyHoldOff***

SNVT Typ: SNVT\_switch, Index 95

Funktion: Eingangsvariable von z.B. Fenster- oder Türkontakt zur Aktivierung der Energiesparfunktion. Mit  $nviEnergyHoldOff = 100.0$  1 wird die Funktion aktiviert und die Stellgrößen Heizen/Kühlen werden auf ihre Minimalwerte zurückgesetzt. Bei aktiver Energiesparfunktion wird die Frostschutzfunktion eingeschaltet (siehe UCPTspAntiFreez). Nach Deaktivierung der Energy-Hold-Off Funktion wird die Temperaturregelung neu gestartet.

### ***nviDewPtSensor***

SNVT Typ: SNVT\_switch, Index 95

Funktion: Eingangsvariable zur Auswertung eines Kondensationswächters im Betriebsmodus Kühlen. Mit  $nviDewPtSensor = 100.0$  1 wird die Stellgröße Kühlen auf ihren Minimalwert zurückgesetzt. Nach Deaktivierung dieser Funktion wird die Temperaturregelung neu gestartet.

## ***Ausgangsvariablen Space Comfort Controller Object***

### ***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 von **SCPTmaxSendTime** und 1,5s- 4s nach Reset.

### ***nvoUnitStatus***

SNVT Typ: SNVT\_hvac\_status, Index 112

Funktion: Ausgangsvariable für den Betriebsstatus und die Stellgrößen Heizen/Kühlen des Reglers.

.mode =	HVAC_HEAT	==>	Heizen
	HVAC_COOL	==>	Kühlen
	HVAC_OFF	==>	Regelung ausgeschaltet
.heat_output_primary	0...100 %	==>	Stellgröße Heizen
.cool_output_primary	0...100 %	==>	Stellgröße Kühlen

### ***nvoHeatPrimary***

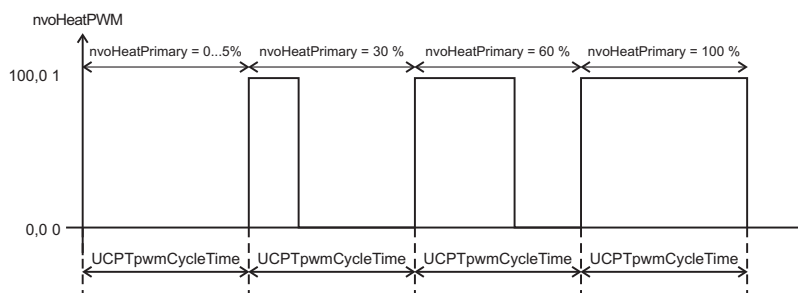
SNVT Typ: SNVT\_lev\_percent, Index 81

Funktion: Ausgangsvariable mit der Stellgröße des PID-Reglers für Heizen zur Ansteuerung eines stetigen Stellantriebs. Die Datenausgabe erfolgt in Abhängigkeit von **SCPTmaxSendTime** und 1,5s- 4s nach Reset.

### ***nvoHeatPWM***

SNVT Typ: SNVT\_switch, Index 95

Funktion: Ausgangsvariable mit der Stellgröße des PID-Reglers für Heizen zur pulsweitenmodulierten Ansteuerung eines thermischen Zweipunkt-Stellantriebs. Die Datenausgabe erfolgt sofort bei einem anstehenden Schaltbefehl und ansonsten in Abhängigkeit von **SCPTmaxSendTime** und 1,5s- 4s nach Reset.





**nvoCoolPrimary**

SNVT Typ: SNVT\_lev\_percent, Index 81

Funktion: Ausgangsvariable mit der Stellgrösse des PID-Reglers für Kühlen. Die Datenausgabe erfolgt analog zu *nvoHeatPrimary*.

**nvoCoolPWM**

SNVT Typ: SNVT\_switch, Index 95

Funktion: Ausgangsvariable mit der Stellgrösse des PID-Reglers für Heizen zur pulswertenmodulierten Ansteuerung eines thermischen Zweipunkt-Stellantriebs. Die Datenausgabe erfolgt analog zu *nvoHeatPWM*.

**nvoSetptEffect**

SNVT Typ: SNVT\_temp\_p, Index 105

Funktion: Die Ausgangsvariable sendet den vom Regelalgorithmus verwendeten Sollwert. Die Ausgabe ist vom Betriebsmodus des Reglers abhängig :

nvUnitStatus.mode = HVAC\_HEAT ==> nvoSetptEffect = Sollwert Heizen

nvUnitStatus.mode = HVAC\_Cool ==> nvoSetptEffect = Sollwert Kühlen

Der effektive Sollwert wird in Abhängigkeit von *nviSetpoint*, *nviOccManCmd*, *nviOccSensor*, *SCPTsetPnts* und *nviSetptOffset* berechnet (siehe Tabelle 1). Die Datenausgabe erfolgt in Abhängigkeit von *SCPTmaxSendTime*, bei Wertänderungen und 1,5s- 4s nach Reset.

**nvoSetptOffset**

SNVT Typ: SNVT\_temp\_p, Index 105

Funktion: Ausgangsvariable für die Sollwertkorrektur, die über *nviSetptOffset* vorgegeben werden kann. Die Datenausgabe erfolgt analog zu *nvoSetptEffect*.

**nvoOccupEffect**

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 1,5s- 4s nach Reset.

**nvoEnergyHoldOff**

SNVT Typ: SNVT\_switch, Index 95

Funktion: Ausgangsvariable zur Statusanzeige der Energiesparfunktion.

nvoEnergyHoldOff = 0.0 0 ==> Fensterkontakt nicht aktiv

nvoEnergyHoldOff = 100.0 1 ==> Fensterkontakt aktiv

Die Datenausgabe erfolgt in Abhängigkeit von *SCPTmaxSendTime*, bei Wertänderungen und 1,5s- 4s nach Reset.

**nvoDewPtSensor**

SNVT Typ: SNVT\_switch, Index 95

Funktion: Ausgangsvariable zur Statusanzeige des Kondensationswächters.

nvoDewPtSensor = 0.0 0 ==> Kondensationswächer nicht aktiv

nvoDewPtSensor = 100.0 1 ==> Kondensationswächer aktiv

Die Datenausgabe erfolgt in Abhängigkeit von *SCPTmaxSendTime*, bei Wertänderungen und 1,5s- 4s nach Reset.

**Konfigurationsparameter Space Comfort Controller Object - Allgemein:****SCPT HVAC Type**

Index: 169, SNVT\_hvac\_type  
 Funktion: Konfigurationsparameter zur Kennzeichnung des Reglertyps.  
 Eingestellter Wert: nciHvacType = HVT\_GENERIC

**SCPT HVAC Mode**

Index: 74, SNVT\_hvac\_mode  
 Funktion: Der Konfigurationsparameter bestimmt den Initialisierungszustand der Eingangsvariablen *nviApplicMode* und damit auch die Startkonfiguration des Temperaturreglers. Voreingestellter Wert: HVAC\_AUTO

**SCPT maxSendTime**

Index: 49, SNVT\_time\_sec  
 Funktion: Der Konfigurationsparameter definiert die Intervallzeit zur Berechnung neuer Stellgrößen für die Temperaturregelung und den Sendezeitpunkt der Ausgangsvariablen. Mit Eingabewerten = 0 wird die Datenausgabe deaktiviert. (Voreingestellter Wert: 30 s)

**SCPT tempOffset**

Index: 272, SNVT\_temp\_diff\_p  
 Funktion: Offset für den Temperaturwert. Mit diesem Parameter ist eine softwareseitige Kalibrierung möglich.

**SCPT numValves**

Index: 59, SNVT\_count  
 Funktion: Der Konfigurationsparameter dient zur Auswahl von 2-Rohr- oder 4-Rohr-Systemen. Wird ein 2-Rohr-System betrieben (1 Ventil), dann erhalten die Ausgangsvariablen mit den Stellgrößen für Heizen und Kühlen die gleichen Werte.

**SCPT numValves = 1: ==> 2-Rohr-System**

Modus Heizen: nvoHeatPrimary = nvoCoolPrimary = Stellgröße Heizen

Modus Kühlen: nvoHeatPrimary = nvoCoolPrimary = Stellgröße Kühlen

**SCPT numValves = 2: ==> 4-Rohr-System (Standardwert)**

Modus Heizen: nvoHeatPrimary = Stellgröße Heizen

Modus Kühlen: nvoCoolPrimary = Stellgröße Kühlen

**UCPT PWM CycleTime**

Index: 35, SNVT\_time\_min  
 Funktion: Der Konfigurationsparameter bestimmt die Zykluszeit zur pulsweitenmodulierten Ansteuerung der Stellantriebe mit *nvoHeatPWM* und *nvoCoolPWM*. (Voreingestellter Wert: 15 min)

**Konfigurationsparameter Space Comfort Controller Object - Sollwert:****SCPT setPnts**

Index: 60, SNVT\_temp\_setpt  
 Funktion: Konfigurationsparameter zur Vorgabe der Sollwerte für Heizen und Kühlen in Abhängigkeit der Raumbelastung. Mit *nviSetpoint* können die Werte bei *nvoOccupEffect* = OCCUPIED bzw. STANDBY überschrieben werden. Bei *nvoOccupEffect* = UNOCCUPIED wird *nviSetpoint* allerdings nicht berücksichtigt.

Voreingestellte Werte:	.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

**UCPT spAntiFreez**

Index: 18, SNVT\_temp\_p  
 Funktion: Sollwert für Heizen zur Frostschutzfunktion bei geöffnetem Fensterkontakt, d.h. bei aktiver Energiesparfunktion. (Voreingestellter Wert: 10 °C)

**Konfigurationsparameter Space Comfort Controller Object - PID-Regler Heizen:****UCPTheatXp**

Index: 19, SNVT\_temp\_p

Funktion: Parameter zur Einstellung des Proportionalbereichs. Mit UCPTheatXp = 0 wird der Regler für Heizen deaktiviert.  
(Voreingestellter Wert: 4 K, Wertebereich: 0-10 K)

**UCPTheatTn**

Index: 20, SNVT\_time\_min

Funktion: Parameter zur Einstellung der Nachstellzeit des I-Anteils (Abtastzeit Ta = SCPTmaxSendTime).  
Mit Eingabewerten = 0 wird der I-Anteil deaktiviert. (Voreingestellter Wert: 100 min)

**UCPTheatTv**

Index: 21, SNVT\_time\_min

Funktion: Parameter zur Einstellung der Vorhaltezeit des D-Anteils (Abtastzeit Ta = SCPTmaxSendTime).  
Mit Eingabewerten = 0 wird der D-Anteil deaktiviert. (Voreingestellter Wert: 0 min)

**UCPTheatMinNight**

Index: 26, SNVT\_lev\_percent

Funktion: Stellgrößenbeschränkung nach Unten im Betriebsmodus UNOCCUPIED.  
(Voreingestellter Wert: 0 %)

**UCPTheatMinDay**

Index: 27, SNVT\_lev\_percent

Funktion: Stellgrößenbeschränkung nach Unten in den Betriebsmodi OCCUPIED und STANDBY.  
(Voreingestellter Wert: 0 %)

**UCPTheatMax**

Index: 28, SNVT\_lev\_percent

Funktion: Stellgrößenbeschränkung nach Oben. (Voreingestellter Wert: 100 %)

**Konfigurationsparameter Space Comfort Controller Object - PID-Regler Kühlen:****UCPTcoolXp**

Index: 22, SNVT\_temp\_p

Funktion: Parameter zur Einstellung des Proportionalbereichs. Mit UCPTcoolXp = 0 wird der Regler für Heizen deaktiviert.  
(Voreingestellter Wert: 4 K, Wertebereich: 0-10 K)

**UCPTcoolTn**

Index: 23, SNVT\_time\_min

Funktion: Parameter zur Einstellung der Nachstellzeit des I-Anteils (Abtastzeit Ta = SCPTmaxSendTime).  
Mit Eingabewerten = 0 wird der I-Anteil deaktiviert. (Voreingestellter Wert: 100 min)

**UCPTcoolTv**

Index: 24, SNVT\_time\_min

Funktion: Parameter zur Einstellung der Vorhaltezeit des D-Anteils (Abtastzeit Ta = SCPTmaxSendTime).  
Mit Eingabewerten = 0 wird der D-Anteil deaktiviert. (Voreingestellter Wert: 0 min)

**UCPTcoolMinNight**

Index: 29, SNVT\_lev\_percent

Funktion: Stellgrößenbeschränkung nach Unten im Betriebsmodus UNOCCUPIED.  
(Voreingestellter Wert: 0 %)

**UCPTcoolMinDay**

Index: 30, SNVT\_lev\_percent

Funktion: Stellgrößenbeschränkung nach Unten in den Betriebsmodi OCCUPIED und STANDBY.  
(Voreingestellter Wert: 0 %)

**UCPTcoolMax**

Index: 31, SNVT\_lev\_percent

Funktion: Stellgrößenbeschränkung nach Oben. (Voreingestellter Wert: 100 %)

## Digital Input Object

Der Zustand des potentialfreien digitalen Eingangs wird erfasst und je nach Konfiguration (UCPTdiConfig) über die Ausgangsvariablen vom Typ SNVT\_switch und SNVT\_setting ausgegeben, wobei mit SNVT\_switch ein absoluter Beleuchtungswert zur manuellen Übersteuerung gesendet wird und mit SNVT\_setting der Occupancy Controller oder der Constant Light Controller aktiviert bzw. deaktiviert werden kann.

### Eingangsvariablen Digital Input Object:

#### nviSwSwitchFb

SNVT Typ: SNVT\_switch, Index 95

Funktion: Eingangsvariablen für den aktuellen Zustand der mit nvoSwSwitch angesteuerten Beleuchtungsgruppen.

### Ausgangsvariablen Digital Input Object:

#### nvoSwSwitch

SNVT Typ: SNVT\_switch, Index 95

Funktion: Die Ausgangsvariablen senden je nach Konfiguration über **UCPTdiConfig** den aktuellen Schaltzustand des Digitaleingangs (aktiv/nicht aktiv) oder Werte zur manuellen Beleuchtungssteuerung.

##### Standard I/O:

Potentialfreier Kontakt **aktiv** ==> nvoSwSwitch.state = 1  
nvoSwSwitch.value = SCPTmaxOut

Potentialfreier Kontakt **nicht aktiv** ==> nvoSwSwitch = 0.0 0

##### Toggle:

Wechsel **nicht aktiv** ==> **aktiv** ==> Jede Tastbetätigung bewirkt ein Umschalten der Variablen zwischen EIN und AUS.

Beleuchtung EIN nvoSwSwitch.state = 1  
nvoSwSwitch.value = SCPTmaxOut

Beleuchtung AUS nvoSwSwitch1/2 = 0.0 0

##### Dimmen:

Wechsel **nicht aktiv** ==> **aktiv** ==> Kurze Tastbetätigungen (< 1 s) führen zum Umschalten des aktuellen Beleuchtungszustandes. Mit längeren Tastbetätigungen (> 1 s) wird die Dimm-Funktion aktiviert, d.h. ausgehend vom aktuellen Beleuchtungszustand wird der .value-wert der Variablen in **SCPTstepValue** - Schritten erhöht oder verringert, so lange wie die Taste gedrückt gehalten wird.

Beleuchtung EIN (max.) nvoSwSwitch.state = 1  
nvoSwSwitch.value=SCPTmaxOut

Beleuchtung EIN (50%) nvoSwSwitch1/2 = 50.0 1

Beleuchtung AUS nvoSwSwitch1/2 = 0.0 0

##### Befehl Automatik:

Wechsel **nicht aktiv** ==> **aktiv** ==> Bei Tastbetätigung wird die Ausgangsvariable nvoSwSwitch mit dem Wert 0.0 -1 gesendet. Damit kann ein angeschlossener Lichtregler in den Automatikmodus geschaltet werden.

Die Ausgangsvariablen werden nach Wechsel des Ausgabewertes, nach Ablauf der Heartbeat-Zeit (SCPTmaxSendTime) und 1,5s- 4s nach Modul-Reset ausgegeben.

#### nvoSwSetting

SNVT Typ: SNVT\_setting, Index 117

Funktion: Ausgangsvariable zur manuellen Steuerung des Betriebszustandes eines Lichtreglers. Es besteht die Möglichkeit den Regler ein- oder auszuschalten und den Sollwert zu verändern. Die Funktion ist ebenfalls über **UCPTdiConfig** konfigurierbar.

##### Standard I/O:

Potentialfreier Kontakt **aktiv** ==> nvoSwSetting.function = SET\_ON Regler EIN

Potentialfreier Kontakt **nicht aktiv** ==> nvoSwSetting.function = SET\_OFF Regler AUS

**Toggle:**

Wechsel *nicht aktiv* ==> *aktiv* ==> Jede Tastbetätigung bewirkt ein Umschalten der Variablen zwischen den Werten

nvoSwSetting.function = SET\_ON            Regler EIN  
nvoSwSetting.function = SET\_OFF          Regler AUS

**Dimmen:**

Wechsel offen ==> geschlossen ==> Kurze Tastbetätigungen (< 1 s) führen zum Umschalten zwischen SET\_ON und SET\_OFF. Mit längeren Tastbetätigungen (> 1 s) wird die Dimm-Funktion aktiviert und der Sollwert des Reglers wird um **SCPTstepValue** verändert (Sendeintervall 400 ms):

nvoSwSetting.function = SET\_UP / SET\_DOWN  
nvoSwSetting.setting = nciSwStepValue

**Konfigurationsparameter Digital Input Object:****SCPTmaxOut**

SCPT Index: 93, SNVT\_lev\_cont

Funktion: Konfigurationsparameter zur Begrenzung des Ausgabewertes der Ausgangsvariablen nvoSwSwitch.value. (Voreingestellter Wert: 100 %)

**SCPTmaxSendTime**

SCPT Index: 49, SNVT\_time\_sec

Funktion: Heartbeatintervall. Nach Ablauf der Zeit SCPTmaxSendTime wird der digitale Eingang abgefragt und die Ausgangsvariablen aktualisiert.

Mit Eingabewerten = 0 wird die Heartbeatfunktion deaktiviert. ( Voreingestellter Wert: 0 )

**SCPTstepValue**

SCPT Index: 92, SNVT\_lev\_cont

Funktion: Mit **SCPTstepValue** wird die Schrittweite der Ausgangsvariablen **nvoSwSwitch.value** definiert, mit der die Werte bei Dimmfunktion verändert werden. Bei Verwendung von SNVT\_setting bestimmt **SCPTstepValue** den Wert von nvoSettingSW.setting. (Voreingestellter Wert: 5 %)

**UCPTdiConfig**

UCPT Index: 44, UNVT\_str\_hex4

Funktion: Über **UCPTdiConfig** können die digitalen Eingänge für die Funktionen Standard I/O, Toggle, Dimmen oder „Befehl Automatik“ konfiguriert werden. (Voreingestellter Wert: 0,0,0,0)

Konfiguration des digitalen Eingangs mit UCPTdiConfig			
Funktion		Öffner / Schließer	
Byte[0]	Beschreibung	Byte[1]	Beschreibung
0	aktiv / nicht aktiv	0	Schließer-Kontakt
1	Licht Toggle	1	Öffner-Kontakt
2	Licht Toggle mit Dimmen		
3	Befehl Automatik (= 0.0 –1)		

**Allgemeine Hinweise:****Service Pin Message**

Solange sich das Gerät im Auslieferungszustand befindet (unconfigured), lässt sich die Service Pin Message auch ohne Betätigung des Service Pins generieren. Damit ist es möglich, ein eingebautes und angeschlossenes Gerät komfortabel zu kommissionieren.

Erkennt der Helligkeitsfühler im Zustand unconfigured 3 x hintereinander einen Wechsel Dunkel ==> Hell, dann wird die Service Pin Message gesendet. Die Grenze Dunkel / Hell liegt bei 800 Lux. Der Zustandswechsel Dunkel ==> Hell lässt sich mit einer Taschenlampe erzeugen, wobei die Impulsdauer für Hell bzw. Dunkel jeweils ca. 3 s beträgt.