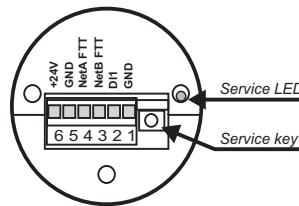
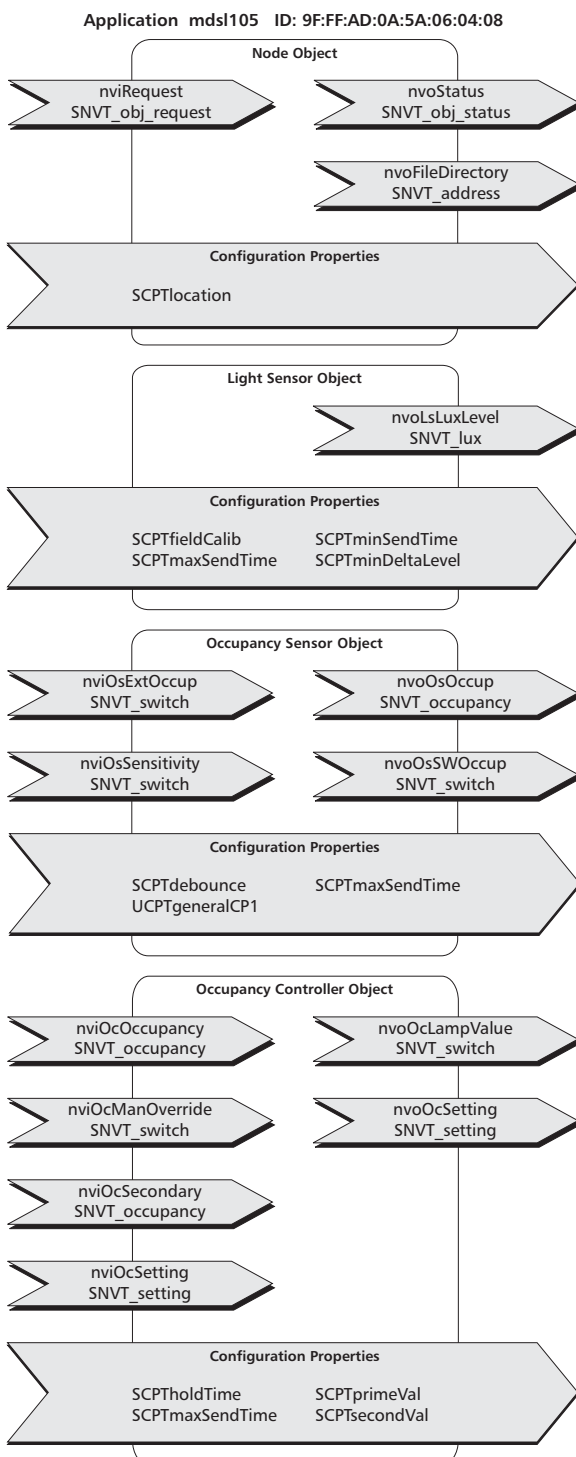


Software Application mds1105 (Sensing Elements, Constant Light Controller)

For sensors type MDS-LON1 (since June 2005)

The ceiling multi-sensor type MDS is designed for occupancy detection, light measuring (0-1000 lux) and temperature detection in room and office spaces. Additionally, it has a potential-free digital input, e.g. for the connection of a light switch.

The function profiles prescribed by the LonMark® are considered: function profile **1010** „Light Sensor“, **1060** „Occupancy Sensor“, **3071** „Occupancy Controller“, **3050** „Constant Light Controller“ and **3200** „Switch“. The application has two identical objects 3050, so that two lighting groups can be controlled in a room independently of each other. The application uses standard network variable types (SNVT) and standard configuration property types (SCPT). For extended adjustment options, user defined configuration property types (UCPT) are used. The UCPTs used are defined in the **Thermokon Device Resource Files** from **Version 1.6** or higher and should be installed on the PC before making up any device defaults by the installation tool.

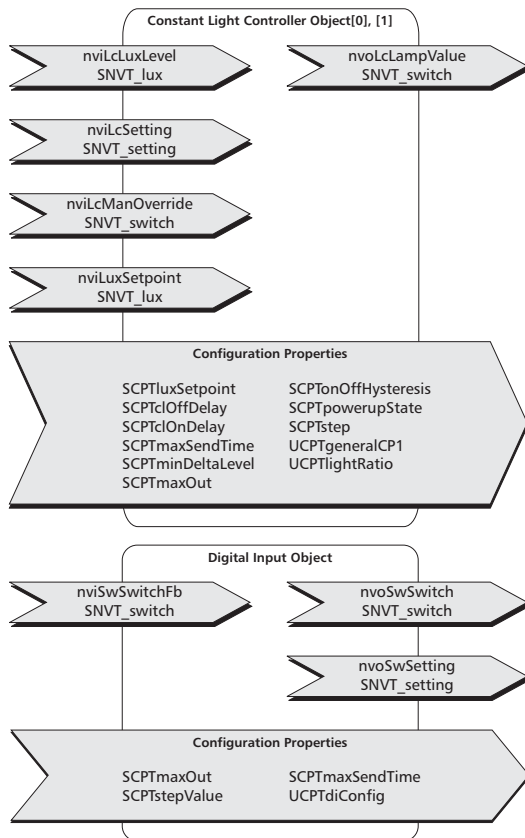


Light Sensor: The light value measured is output by the variable **nvoLsLuxLevel**. For the calibration of a light sensor, the exact light intensity can be determined by an external luxmeter and can be entered by the parameter **SCPTfieldCalib**. The reflection factor is automatically calculated and both, the measuring value and the measuring range end value are corrected, accordingly.

Occupancy Sensor: The current room occupancy is output by variables of the type SNVT_occupancy and SNVT_switch. The setting back of the output variables after detected movement is made delayed (adjustable via **SCPTdebounce**).

By means of the input/output variables of type SNVT_switch, the occupancy detector offers additionally the possibility to connect several occupancy sensors or to control a lighting directly depending on the occupancy. By **nviOsSensitivity** the sensitivity of the occupancy sensor can be reduced. The integrated occupancy sensor-LED always lights up for the time of the initialization phase. By **UCPTgeneralCP1**, it is adjustable whether the LED should shine with detected movement.

Occupancy Controller: The occupancy controller can be used as a light switch depending on movement (by **nvoOcLampValue**) or for switching-on/-off a connected constant light controller (by **nvoOcSetting**). The setting back of the output variable after detected occupancy is made delayed (adjustable by **SCPTholdTime**). The input variable **nviOcOccupancy** can be connected to the output variable **nvoOsOccup** of the **Occupancy Sensor Objects** (internal occupancy detector). By **nviOcSecondary = OCCUPIED** the lighting is switched-on to the value **SCPTsecondVal**. By **nviOcSetting** the controller can be activated respectively deactivated and by **nviOcManOverride** the possibility is given to override the controller externally.



Constant Light Controller Objects [0] and [1]:

Two identical objects for light control respectively light regulation. By **UCPTgeneralCP1** the configuration of the controllers is made. They can be used as constant light controller as well as occupancy sensors depending on brightness. Configured as a brightness-dependent occupancy sensor, two lighting groups in one room can be controlled and switched independently of each other by these two objects. The input variable **nviLcLuxLevel** must be connected to the light sensor object and the output variable **nvoLcLampValue** must be connected to an actuator for light control.

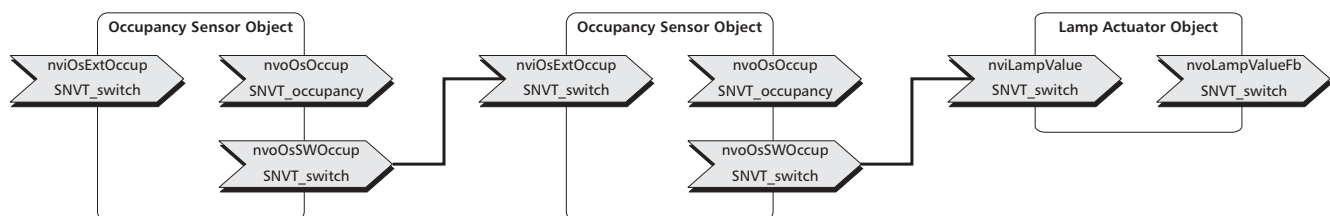
By **nviLcSetting** the controller can be switched-on/-off and the set point can be changed temporarily. The input variable **nviLcManOverride** is designed for manual override of the light value.

Digital Input Object: The inverter state of the potential-free digital input is detected and output via the output variable of type SNVT_switch and SNVT_setting depending on the configuration (UCPTdiConfig). By SNVT_switch an absolute switch value for manual override is sent. By SNVT_setting the occupancy controller or the constant light controller can be activated respectively deactivated. The digital inputs can take over the functions standard I/O, toggle, dim or automatic. By SCPTmaxOut the maximum output value of the SNVT_switch variable can be limited.

Application Examples:

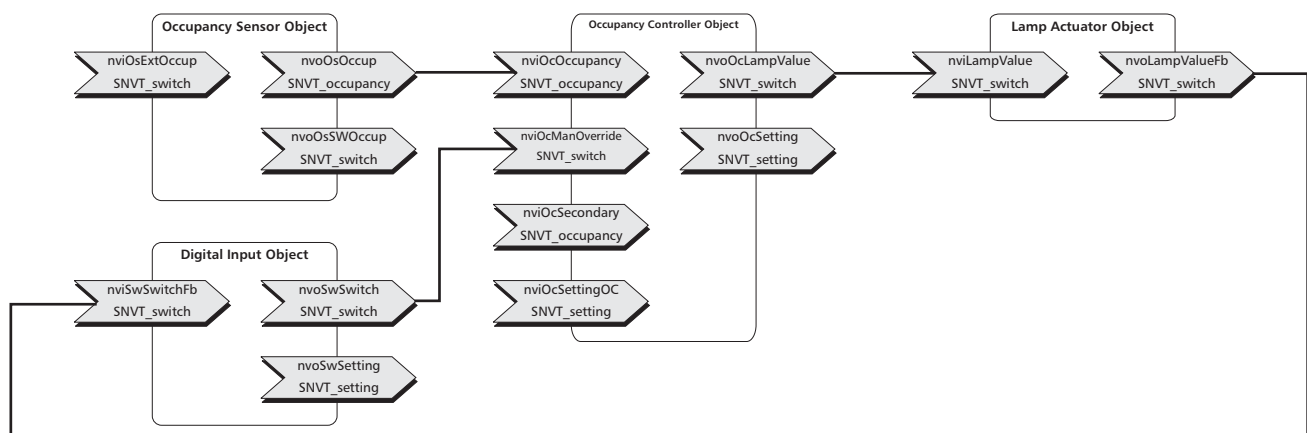
Occupancy Sensor:

Logical OR-circuit link of 2 occupancy sensors and a direct light control



Occupancy Controller:

Switching of the lighting depending on movement by a manual override via keys with toggle function



Constant Light Controller:**Calibration and Installion Notice:**

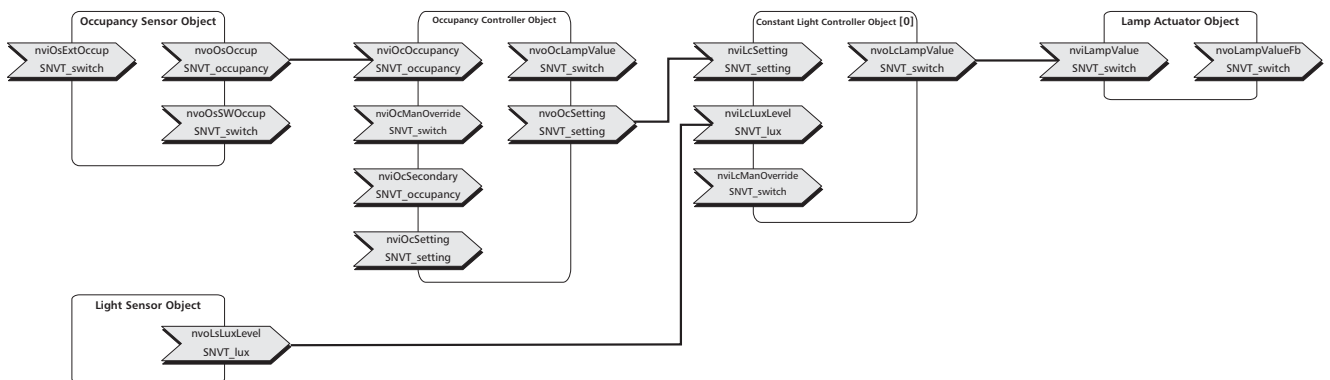
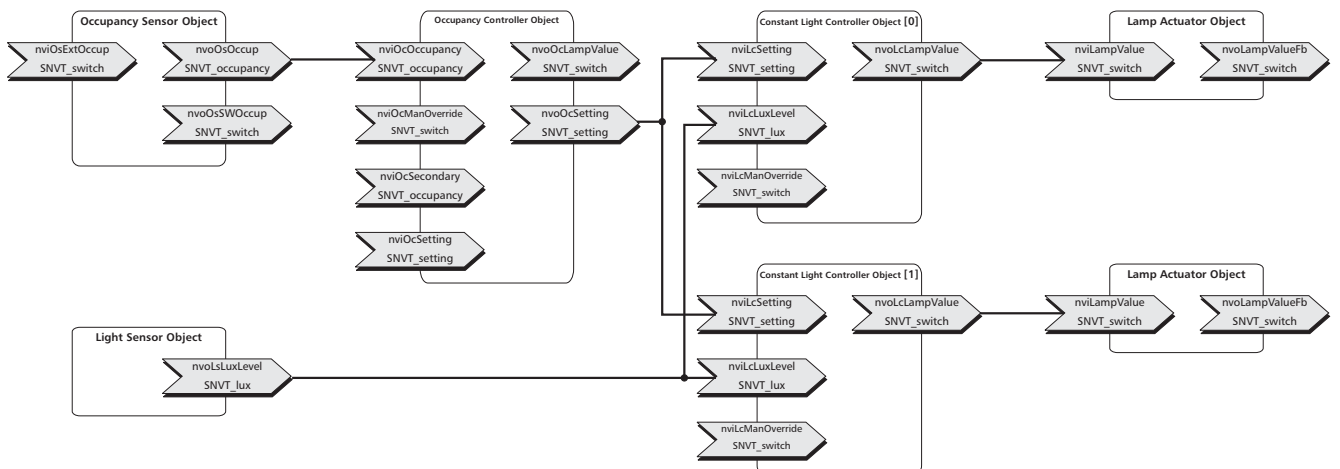
1. Light Sensor: The calibration of the light sensor should be made without any artificial light and in a blind position typical for the room. Thus, the light intensity is measured at the working surface by means of a reference device and entered as a configuration value in the parameter **SCPTfieldCalib**.

2. Constant Light Controller: As the sensitivity of the light sensor compared to artificial light is above all depending on the light source, the location of the sensor and the reflection characteristics of the room, the sensitivity must be determined for installation by the configuration property **UCPTlightRatio**.

UCPTlightRatio.multiplier: Light intensity with 100 % artificial light, measured on the working surface by a reference device.

UCPTlightRatio.divisor: Light intensity with 100 % artificial light, measured by the multi-sensor MDS LON1.

The determinations of the values must be made in a well darkened room, if possible without any daylight. The artificial light should be switched to its maximum value of 100%.

Constant light control of a light group by activation via the occupancy controller**Occupancy sensor depending on brightness for switching two light groups by activation via the occup. controller**

Node Object

The Node Object supervises and controls the functions of the individual objects in a device. The basic functionality required by the LonMark[®] is supported.

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.

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 parameter in the device available to the LON integration tool.

Configuration PProperties Node Object:

SCPTlocation

SCPT Index: 17, SNVT_str_asc

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

Light Sensor Object

The object includes the functions for measuring the light intensity and data output.

Output Variable Light Sensor Object:

nvoLsLuxLevel

SNVT Type: SNVT_lux, Index 79

Function: Output variable for the measured light intensity in lux. Data output is made depending on the configuration parametera *SCPTminSendTime* , *SCPTmaxSendTime* and *SCPTminDeltaLevel* and 1,5s- 4s after reset.

Configuration Properties Light Sensor Object:

SCPTfieldCalib

SCPT Index: 90, SNVT_lux

Function: Configuration property for self-calibration of the light sensor. By means of an external luxmeter the precise light intensity can be determined and input. The reflection factor is automatically calculated and both, measuring value and measuring end value are corrected, accordingly.

!! Calibration should be made without artificial light and in a blind position typical for the room

!! The calibration value is only allowed to be input if the sensor is directly connected to the network,

!! as otherwise the correction value won't be calculated

(Preset value: 0 Lux ==> field calibration deactivated)

SCPTmaxSendTime

SCPT Index: 49, SNVT_time_sec

Function: Heartbeat function. Stipulates the interval time, after which the output variables are sent independently of a value change. By the input value = 0 the heartbeat function is deactivated. (Preset value: 60 sec.)

SCPTminSendTime

SCPT Index: 52, SNVT_time_sec

Function: Heartbeat function. Stipulates the smallest update interval of the output variable. An update is made after expiration of *SCPTminSendTime*, if the light value has changed by at least *SCPTminDeltaLevel*. By input values = 0, the „Minsend“ function is deactivated. (Preset value: 1 sec.)

SCPTminDeltaLevel

SCPT Index: 88, SNVT_lev_cont

Function: If the light intensity has changed by the adjusted value ***SCPTminDeltaLevel*** (% of the measuring range), the new light values are transmitted. The function is depending on the adjustment ***SCPTminSendTime***. (Value range: 0 % - 100 %; preset value: 2,5 %)

Occupancy Sensor Object***Input Variable Occupancy Sensor Object:******nviOsExtOccup***

SNVT Type: SNVT_switch, Index 95

Function: Input variable for external occupancy detector (e.g. logical OR circuit link of several occupancy sensors). By ***nviOsExtOccup*** = 100,0 1 the output variables are set to OC_OCCUPIED respectively 100,0 1. With other values, the output variables are set back after expiration of the delay time ***SCPTdebounce***. The internal IR-occupancy sensor is logical OR-circuit linked to the controller via ***nviOsExtOccup***.

nviOsSensitivity

SNVT Type: SNVT_switch, Index 95

Function: By ***nviOsSensitivity*** = 0,0 0 the sensitivity of the occupancy detector can be reduced. Initialization value after reset: ***nviOsSensitivity*** = 100,0 1, i.e. high sensitivity.

Output Variables Occupancy Sensor Object:***nvoOsOccup***

SNVT Type: SNVT_occupancy, Index 109

Function: Output variable occupancy detector. It is determined as soon as an internal or external movement is recognized. The setting back is made after expiration of the delay time ***SCPTdebounce***. Data transmission is made depending on the configuration properties ***SCPTdebounce*** and ***SCPTmaxSendTime***. **Module-Reset:** For the first 60 sec. after reset (initialization phase of the occupancy detector) no data transmission is made and ***nvoOccup*** receives the value OC_UNOCCUPIED.

nvoOsSWOccup

SNVT Type: SNVT_switch, Index 95

Function: Output variable occupancy message. Is sent parallel with ***nvoOsOccup***. This variable can be evaluated as an „external occupancy message“ by another occupancy detector or can directly control a lighting group.

Configuration Properties Occupancy Sensor Object:***SCPTmaxSendTime***

SCPT Index: 49, SNVT_time_sec

Function: Heartbeat function. Stipulates the interval time, after which the output variables are sent independently of a result change. By the input value = 0, the heartbeat function is deactivated. (Preset value: 120 sec.)

SCPTdebounce

SCPT Index: 139, SNVT_time_sec

Function: Time delay for setting back the output variables after recognized movement. The delay timer is started after status change „movement ==> no movement“. (Preset value: 0 sec.)

UCPTgeneralCP1

UCPT Index: 7, SNVT_state

Function: The integrated occupancy detector LED always lights up for the time of the initialization phase. If the LED should shine with detected movement, this can be adjusted by ***UCPTgeneralCP1***.

UCPTgeneralCP1.bit0 = 0 ==> LED only shines in the initialization phase of the device

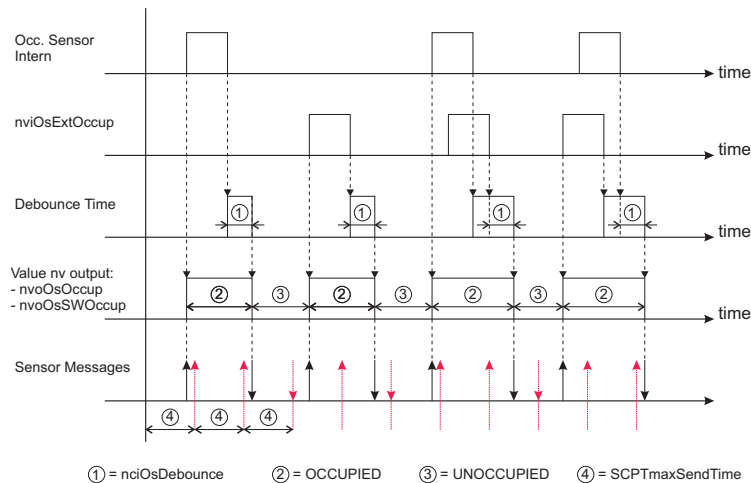
UCPTgeneralCP1.bit0 = 1 ==> LED blinks additionally with detected movement (default)

UCPTgeneralCP1 (continue)

Function: Parameter to configure the output variable nvoOcSetting of the Occupancy-Controller-Object.

UCPTgeneralCP1.bit15 = 0 ==> nvoOcSetting.function transmit SET_ON and SET_OFF (default)

UCPTgeneralCP1.bit15 = 1 ==> nvoOcSetting.function transmit only SET_OFF

Function Diagram Occupancy Sensor:**Occupancy Controller Object**

By **nvoOcLampValue** the occupancy controller can be used as light switch depending on movements or by **nvoOcSetting** for switching-on/-off a connected constant light controller.

Input Variable Occupancy Controller Object:**nviOcOccupancy**

SNVT Type: SNVT_occupancy, Index 109

Function: The input variable **nviOcOccupancy** makes the current room occupancy available to the controller and is connected with the output variable **nvoOsOccup** of the occupancy sensor. (Initialization value after reset: OC_NUL)

nviOcManOverride

SNVT Type: SNVT_switch, Index 95

Function: Input variable for manual control of the lighting, independently of the variable value **nviOcSetting**. An update of **nviOcManOverride** locks the controller and the output variable **nvoOcLampValue** takes over the values of **nviOcManOverride**.

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

nviOcManOverride.state = 1 ==> nvoOcLampValue = nviOcManOverride

If the occupancy controller should be deactivated by **nviOcManOverride**, the controller is set back in the automatic mode after receipt of UNOCCUPIED to **nviOcOccupancy** and after expiration of the delay time SCPTHoldTime. (Initialization value after reset: 0.0 -1)

nviOcSecondary

SNVT Type: SNVT_occupancy, Index 109

Function: Input variable of a neighbouring occupancy detector with the current room occupancy of a neighbouring area. (Initialization value after reset: OC_NUL)

nviOcSetting

SNVT Type: SNVT_setting, Index 117

Function: The input variable **nviOcSetting** activates respectively deactivates the controller.

Initialisation status after reset: nviSettingOC.function = SET_ON

nviOcSetting.function = SET_OFF ==> Controller = OFF; nvoOcLampValue = 0.0 0 (lighting OFF)

nviOcSetting.function = SET_ON ==> Controller = ON;

Output Variable Occupancy Controller Object:**nvoOcLampValue**

SNVT Type: SNVT_switch, Index 95

Function: Output variable for control of lighting (see function diagram occupancy controller)

nvoOcLampValue.state = 0 ==> Lighting OFF

nvoOcLampValue.state = 1 ==> Lighting ON

nvoOcLampValue.value = light intensity (0 - 100 %)

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

nvoOcSetting

SNVT Type: SNVT_setting, Index 117

Function: Output variable for control of a down-stream controller, e.g. constant light controller (see function diagram occupancy controller). Data output is made analog to **nvoOcLampValue**.

nviOccupancy or nviSecondary = OCCUPIED ==> nvoSettingOC.function = SET_ON

nviOccupancy and nviSecondary = UNOCCUPIED ==> nvoSettingOC.function = SET_OFF

Configuration Properties Occupancy Controller Object:**SCPTholdTime**

SCPT Index: 91, SNVT_time_sec

Function: Time delay for setting back the output variables **nvoOcLampValue** and **nvoOcSetting** after **nviOcOccupancy** and **nviOcSecondary** have taken over the status UNOCCUPIED. The delay timer is started after the status change „OCCUPIED ==> UNOCCUPIED“.

(Preset value: 600,0 sec = 10 min)

SCPTprimeVal

SCPT Index: 155, SNVT_switch

Function: The configuration properties **SCPTprimeVal** define the output value of **nvoLampValueOC** if nviOcOccupancy = OC_OCCUPIED. (Preset value: 100.0 1)

SCPTsecondVal

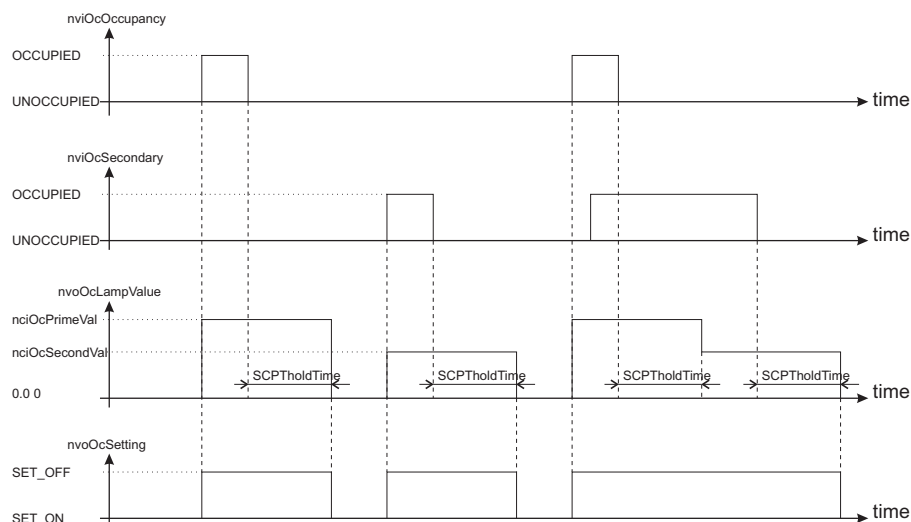
SCPT Index: 156, SNVT_switch

Function: The configuration parameter **SCPTsecondVal** defines the output value **nvoLampValueOC** if nviOccupancy = UNOCCUPIED and nviSecondary = OCCUPIED. (Preset value: 0.0 0)

SCPTmaxSendTime

SCPT Index: 49, SNVT_time_sec

Function: Heartbeat function. Stipulates the interval time after which the output variables are sent independently of a result change. By the input value = 0 the heartbeat function is deactivated. (Preset value: 120 sec)

Function Diagram Occupancy Controller:

Constant Light Controller Object [0], [1]

Two identical objects for the control of the light intensity on a prescribed set point. By *UCPTgeneralCP1* the function constant light controller can be converted into the function of an occupancy detector depending on brightness for light control (switching-on/-off the light depending on the room occupancy and the brightness).

!! Configured as an occupancy sensor depending on brightness, both light controller objects can control and switch !! along with the light sensor of the MDS two lighting groups in a room independently of each other.
!! Configured as a constant light controller the second light controller object requires an additional light sensor !! for control.

Input Variables Constant Light Controller Object:

nviLcLuxLevel[0], [1]

SNVT Type: SNVT_lux, Index 79

Function: The input variable includes the current light intensity in the room and is connected to the output variable *nvoLsLuxLevel* of the light sensor.

nviLcSetting[0], [1]

SNVT Type: SNVT_setting, Index 117

Function: The input variable determines the operating status of the controller (ON or OFF) and can additionally be used for a temporary set point adjustment.

nviLcSetting.function = SET_ON: Controller = ON, i.e. the output quantity for light control (*nvoLcLampValue*) is changed in that way, that the light intensity in the room corresponds to the set point adjusted.

nviLcSetting.function = SET_OFF: Controller = OFF and lighting OFF (*nvoLcLampValue* = 0.0 0)

When having a configuration as a constant light controller (*UCPTgeneralCP1.bit14* = 0 and *UCPTgeneralCP1.bit15* = 0):

nviLcSetting.function = SET_UP: Increasing of the output variable *nvoLcLampValue.value* by the value *nviLcSetting.setting*. The new light value is becoming automatically the new lighting set point.

nviLcSetting.function = SET_DOWN: Reducing of the output variable *nvoLcLampValue.value* by the value *nviLcSetting.setting*. The new light value is becoming automatically the new lighting set point.

By an update to SET_ON the set point is reset to the basic set point SCPTluxSetpoint.

nviLcManOverride[0], [1]

SNVT Type: SNVT_switch, Index 95

Function: Input variable for manual control of the lighting. Initialization value after reset: 0.0 -1

With a configuration as a constant light controller (*UCPTgeneralCP1.bit14* = 0, *UCPTgeneralCP1.bit15* = 0) or as an occupancy sensor depending on the brightness for light control (*UCPTgeneralCP1.bit14* = 0, *UCPTgeneralCP1.bit15* = 1):

An update of *nviLcManOverride* locks the controller and the output variable *nvoLcLampValue* takes over the values of *nviLcManOverride*. By *nviLcManOverride.state* = -1 the controller is activated.

nviLcManOverride.state = -1 ==> Light controller ON

nviLcManOverride.state = 0, 1 und *.value* = 0 - 100 % ==> Light controller OFF

==> *nvoLcLampValue* = *nviLcManOverride*

With a configuration to the switching-off of the lighting depending on the brightness (*UCPTgeneralCP1.bit14* = 1 and *UCPTgeneralCP1.bit15* = 0):

With an update of *nviLcManOverride* = 100.0 1 the lighting is switched-on by *nvoLcLampValue* = 100.0 1. The lighting is switched-off depending on the measured light intensity in the room (see function diagram on page 10).

Automatic reset to value 0.0 -1, only if *UCPTgeneralCP1.bit1* = 1

If *nviLcSetting.function* changes to SET_OFF the output variable *nviLcManOverride* is set to 0.0 -1.

nviLuxSetpoint[0], [1]

SNVT Typ: SNVT_lux, Index 79

Funktion: Input variable for default of a set point for light control. nviLuxSetpoint is initialized by *SCPTluxSetpoint*.**Output Variables Constant Light Controller Object:****nvoLcLampValue[0], [1]**

SNVT Type: SNVT_switch, Index 95

Funktion: Output variable for light control

nvoLcLampValue.state = 0 ==> Lighting OFF

nvoLcLampValue.state = 1 ==> Lighting ON

nvoLcLampValue.value = light intensity (0 - 100 %)

Data output is made depending on the configuration properties *SCPTminDeltaLevel*, *SCPTmaxSendTime* and 1,5s- 4s after reset.**Configuration Properties Constant Light Controller Object:****SCPTluxSetpoint**

SCPT Index: 82, SNVT_lux

Funktion: Initialisation value of nviLuxSetpoint after reset. (Preset value: 500 lux)

SCPTcIOffDelay

SCPT Index: 85, SNVT_time_sec

Funktion: Switch-off delay for lighting (nvoLcLampValue.state = 0).

If the light value exceeds the limit value (*SCPTluxSetpoint* + *SCPTonOffHysteresis*/2) for the time *SCPTcIOffDelay*, the lighting is switched-off. With a configuration as a constant light controller, the automatic switching-off is deactivated by *SCPTonOffHysteresis* = 0. (Preset value: 300,0 sec = 5 min)**SCPTcIONDelay**

SCPT Index: 86, SNVT_time_sec

Funktion: Switching-on delay for lighting (nvoLcLampValue.state = 1).

If the limit value (*SCPTluxSetpoint* - *SCPTonOffHysteresis*/2) is under-run for the time *SCPTcIONDelay*, the lighting is switched-on. (Preset value: 0 sec).**SCPTmaxSendTime**

SCPT Index: 49, SNVT_time_sec

Funktion: Heartbeat function. Stipulates the interval time, after which the output variables are sent independently of a result change.

By the input value = 0, the heartbeat function is deactivated. (Preset value: 300 sec = 5 min)

SCPTminDeltaLevel

SCPT Index: 88, SNVT_lev_cont

Funktion: If the output value changes by the adjusted value *SCPTminDeltaLevel*, *nvoLcLampValue* is sent. (Value range: 0 % - 100 %; preset value: 0,5 %)**SCPTonOffHysteresis**

SCPT Index: 84, SNVT_lev_cont

Funktion: Relative hysteresis value (% of *SCPTluxSetpoint*) for calculation of the switching thresholds on which the lighting is switched-on respectively switched-off depending on the delay times *SCPTcIONDelay* and *SCPTcIOffDelay*. With a configuration as a constant light controller, the automatic switching-off is deactivated by *SCPTonOffHysteresis* = 0.ten (preset value 0 %).Automatic switching-on: $nviLcLuxLevel < SCPTluxSetpoint - SCPTonOffHysteresis/2$ Automatic switching-off: $nvoLcLampValue.value = 0$

and

 $nviLcLuxLevel > SCPTluxSetpoint + SCPTonOffHysteresis/2$ **SCPTpowerupState**

SCPT Index: 87, SNVT_setting

Funktion: Initialization value for the operating mode of the controller after reset. (Preset value: {SET_OFF,0,0})

UCPTlightRatio

UCPT Index: 11, SNVT_muldiv

Function: Configuration property to determine the sensitivity of the light sensor compared to artificial light.
 Preset value: 1000, 100

UCPTlightRatio.multiplier: Light intensity with 100 % artificial light, measured on the working surface by a reference device.

UCPTlightRatio.divisor: Light intensity with 100 % artificial light, measured by the multi-sensor MDS via *nvoLsLuxLevel*.

**!! The determination of the values must be made in a well darkened room, preferably without daylight
 !! The artificial light should be switched to his maximum value of 100 %.**

UCPTgeneralCP1

UCPT Index: 7, SNVT_state

Function: Configuration property for adjustment of the controller functionality
 (Preset value: UCPTgeneralCP1 = 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0 i.e. constant light control)

UCPTgeneralCP1 .bit14	UCPTgeneralCP1 .bit15	Function
0	0	Constant light control(Constant LC) UCPTgeneralCP1.bit0 =0: Start value nvoLcLampValue=SCPTmaxOut UCPTgeneralCP1.bit0 =1: Start value nvoLcLampValue=0%
0	1	Occupancy detector for light control depending on the brightness (ON/OFF LC) Function diagram:
1	0	Switching-off the lighting depending on the brightness (OFF LC). The switching-on of the lighting is made by nviLcManOverride = 100 1 Function diagram:

UCPTgeneralCP1 (continue)

Additional functionality with UCPTgeneralCP1.bit1 (default value = 0)

UCPTgeneralCP1.bit1 = 1: nviLcManOverride is set to 0.0 -1 id nviLcSetting.function changes to SET_OFF.

UCPTgeneralCP1.bit1 = 0: nviLcManOverride can only be modified by the network.

SCPTstep

SCPT Index: 83, SNVT_lev_cont

Function: Maximum step size by which the output variable *nvoLcLampValue.value* of the light controller is allowed to be changed, to reach the adjusted set point.
(Preset value: 10 %)

SCPTmaxOut

SCPT Index: 93, SNVT_lev_cont

Function: Configuration property for limiting the output value of the output variables *nvoLcLampValue*, if the controller automatically changed the light value. When having an external default by *nviLcManOverride* the output value of *nvoLcLampValue* is not limited. (Preset value: 100 %)

Digital Input Object

The status of the potential-free input is detected and output by the output variables of type SNVT_switch and SNVT_setting depending on the configuration(UCPTdiConfig), whereas an absolute light value for manual override is sent by SNVT_switch and the occupancy controller or the constant light controller can be activated respectively deactivated by SNVT_setting.

Input Variables Digital Input Object:**nviSwSwitchFb**

SNVT Type: SNVT_switch, Index 95

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

Output Variable Digital Input Object:**nvoSwSwitch**

SNVT Type: SNVT_switch, Index 95

Function: Depending on the configuration by *UCPTdiConfig* the output variables are sending the current switching status of the digital inputs (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:

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

Lighting ON nvoSwSwitch.state = 1
nvoSwSwitch.value = SCPTmaxOut
Lighting OFF nvoSwSwitch1/2 = 0.0 0

Dim:

Change *inactive* ==> *active* ==> Short-term button actuations (< 1 s) lead to a toggling of the current light status. By longer button actuations (> 1 s) the dimming function is activated, i.e. based on the current light status, the .value-value of the variables is increased or reduced in *SCPTstepValue* -steps as long as the button is kept pressed.

Lighting ON (max.) nvoSwSwitch.state = 1
nvoSwSwitch.value = SCPTmaxOut
Lighting ON (50%) nvoSwSwitch1/2 = 50.0 1
Lighting OFF nvoSwSwitch1/2 = 0.0 0

nvoSwSwitch (continue)

Automatic Command:

Change *inactive* ==> *active*

==> By button actuation, the output variable nvoSwSwitch is sent with the value 0.0 -1. Thus, a connected light controller can be switched in the automatic mode.

The output variables are sent after change of the output value, after expiration of the heartbeat time (SCPTmaxSendTime) and 1,5s- 4s after module reset.

nvoSwSetting

SNVT Type: SNVT_setting, Index 117

Function: Output variable for manual control of the operating status of a light controller. It is possible to switch the controller on or off and to change the set point. The function is also configurable by *UCPTdiConfig*.

Standard I/O:

Potential-free contact *active*

==> nvoSwSetting.function = SET_ON Controller ON

Potential-free contact *inactive*

==> nvoSwSetting.function = SET_OFF Controller OFF

Toggle:

Change *inactive* ==> *active*

==> Each button actuation results in a toggling of the variables between the values

nvoSwSetting.function = SET_ON Controller ON

nvoSwSetting.function = SET_OFF Controller OFF

Dim:

Change open ==> closed

==> Short-term button actuations (< 1 s) result in a toggling between SET_ON and SET_OFF. By longer button actuations (> 1 s) the dimming function is activated and the set point of the controller is changed by *SCPTstepValue* (sending interval: 400 ms):

nvoSwSetting.function = SET_UP / SET_DOWN

nvoSwSetting.setting = SCPTstepValue

Configuration Property Digital Input Object:

SCPTmaxOut

SCPT Index: 93, SNVT_lev_cont

Function: Configuration parameter for limiting the output value of the output variable *nvoSwSwitch.value*. (Preset value: 100 %)

SCPTmaxSendTime

SCPT Index: 49, SNVT_time_sec

Function: Heartbeat interval. After expiration of the time SCPTmaxSendTime the digital input is inquired and the output variables are updated. By the input values = 0, the heartbeat function is deactivated. (Preset value: 0)

SCPTstepValue

SCPT Index: 92, SNVT_lev_cont

Function: By SCPTstepValue the step size of the output variables nvoSwSwitch.value is defined by which the values with the dimming function are changed. By the use of SNVT_setting, SCPTstepValue determines the value of nvoSettingSW.setting. (Preset value: 5 %)

UCPTdiConfig

UCPT Index: 44, UNVT_str_hex4

Function: By UCPTdiConfig the digital inputs can be configured for the functions standard I/O, toggle, dim or „automatic command“. (Preset value: 0,0,0,0)

Configuration of digital inputs by UCPTdiConfig			
function		open/closed contacts	
Byte[0]	Description	Byte[1]	Description
0	actuated / not actuated	0	closed contact
1	Light Toggle	1	open contact
2	Light Toggle with dimming		
3	Command automatic (= 0.0 –1)		

General Notice:**Wink - Event**

The service LED is controlled and blinks two times.

Service Pin Message

As long as the device is unconfigured, the Service Pin Message can be generated without actuation of the service pin. Thus, it is possible to comfortably commission a built-in and connected device.

If the light sensor recognises 3 times in a row a change from dark ==> bright in the status unconfigured, the service pin message is sent. The limit dark/bright amounts to 800 lux. The status change dark ==>bright can be produced by means of a pocket lamp, whereas the impulse time for bright respectively dark amounts to approx. 3 s.