

DE - Softwarebeschreibung

Technische Änderungen vorbehalten
Stand 11.04.07

29100...

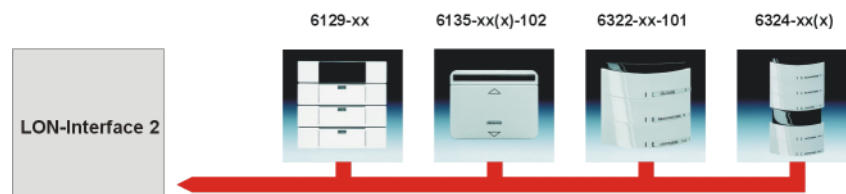


1 Anwendung

Mit dem LON-Interface 2 (bus coupling unit) können Geräte aus dem Busch-Jaeger EIB Programm:

- alpha nea / alpha exclusive: 6135-xx-102
- solo / future / future linear / carat: 6129-xx
- triton: 6322-xx-101, 6324-xx

an das Gebäude-Kommunikationssystem LON der Fa. Echelon angekoppelt werden. Dabei dient das LON-Interface als Schnittstelle zwischen dem LON Bus und des spezifischen Busch-Jaeger Geräts.



Die Vorgaben des LonMark® Funktionsprofile **3200** „Switch“ und **3250** „Scene Panel“ wurden berücksichtigt. Für erweiterte Einstellmöglichkeiten werden benutzerdefinierte Konfigurationsparameter (UCPT) genutzt. Die hier verwendeten UCPTs sind in den Thermokon Device Resource Files ab Version 2.1 oder höher definiert.

2 Geräteübersicht

6135-xx-102, 6129-xx, 6322-xx-101, 6324-xx

- Universal Bedienelement mit IR-Empfang
- Tasterfunktion (Schalten/Dimmen/Jalousie/Rollladen/Wert senden/Lichtszenen)
- Bedienelemente: Tastkontakte links/rechts bzw. oben/unten
- Anzeigeelemente: LED zur Anzeige des Schaltzustandes
- Sondertasten (M1, M2, Alles aus, Memo) der Fernbedienung werden nicht ausgewertet

3 Parametrierung der Wippen

Zum Parametrieren der einzelnen Wippen wird der Konfigurationsparameter *UCPTdiConfig[...]* im Node Object verwendet. Bei der Benutzung des LONMakers kann durch einen Doppelklick auf den Parameter das unten dargestellte Konfigurierungsfenster geöffnet werden.

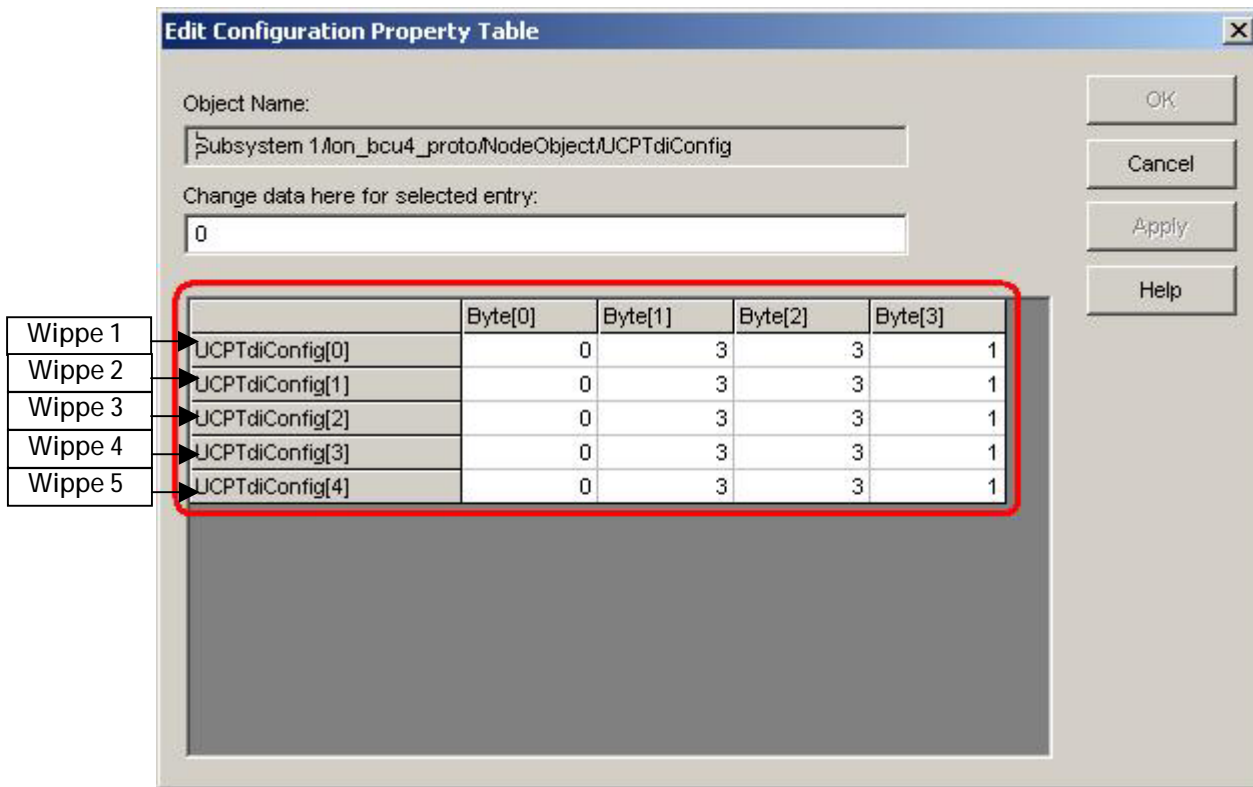


Abbildung 3-1: LONmaker

UCPTdiConfig[0], UCPTdiConfig[1]

Dabei wird die:

- Wippe 1 über UCPTdiConfig[0]
 - Wippe 2 über UCPTdiConfig[1]
 - Wippe 3 über UCPTdiConfig[2]
 - Wippe 4 über UCPTdiConfig[3]
 - Wippe 5 über UCPTdiConfig[4]
- konfiguriert.

Byte[0]

Die Funktionen der Wippe wird über das Byte[0] konfiguriert. Folgende Funktionen sind möglich:

- 0 - Schalten
- 1 – Dimmen
- 2 – Jalousie
- 3 - Rollladen
- 4 – Szene

Beispiel: Die Wippe 1 soll als Dimmfunktion und die Wippe 2 als Jalousie konfiguriert werden.
UCPTdiConfig[0].Byte[0] = 1
UCPTdiConfig[1].Byte[0] = 2

Parametrierung der Wippen

Byte[1], Byte[2]

Über das Byte[1] wird die Funktion der linken und mit Byte[2] die Funktion der rechten Taste der Wippe konfiguriert. Es kann z.B. ausgewählt werden, ob die linke Taste das Licht Aus (Byte[1] = 6) oder Ein (Byte[1] = 4) schalten soll. (siehe auch Kapitel: 4.1.3)

Beispiel: Die linke Taste soll das Licht Ein / heller und die rechte Aus / dunkler schalten / dimmen.
UCPTdiConfig[0].Byte[1] = 3
UCPTdiConfig[0].Byte[2] = 7

Byte[3]

Die LED der Wippe wird im Byte[3] eingestellt. So kann ausgewählt werden, ob die Farbe zwischen rot und grün bei einer Tastenbetätigung geändert werden soll. (siehe auch Kapitel:4.1.3)

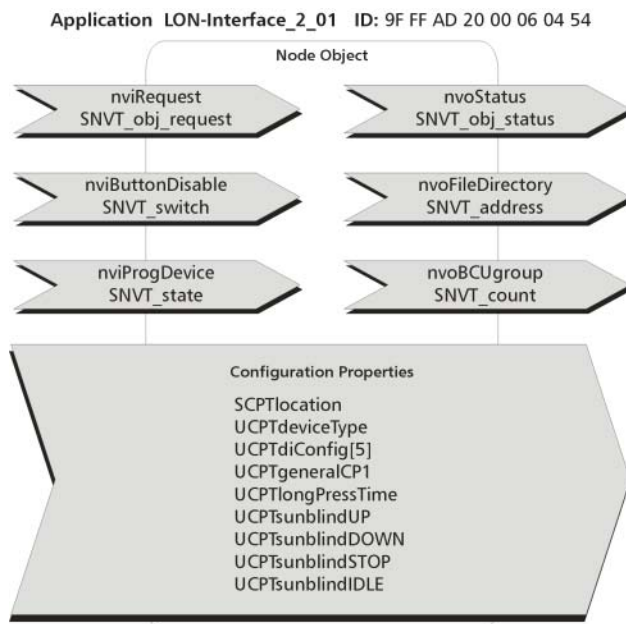
Beispiel: Die LED soll bei einem linken Tastendruck rot und bei einem rechten Tastendruck grün sein.
UCPTdiConfig[0].Byte[3] = 1

Übertragen der Daten

Nachdem Parameter geändert wurden sind, müssen die Parameter zum Bedienelement übertragen werden. Mit setzen von 0 auf 1 von nviProgDevice.bit0 bzw. nach einem Spannungsreset werden diese neuen Einstellungen in das Bedienteil übernommen.

4 Softwarebeschreibung

4.1 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. Nach einer Parametrierung muss *nviProgDevice.bit0* von 0 auf 1 eingestellt werden, damit die neuen Parameter übernommen werden.

Tasterfunktionalität und Gerätevarianten:

Während der Inbetriebnahme muss über den Parameter *UCPTdeviceType* der entsprechend verwendete Gerätetyp (6135-xx-102, 6129-xx, 6322-xx-101, 6324-xx) eingestellt werden. Den Wippen können unterschiedliche Funktionen z.B. Beleuchtungssteuerung oder Szenenaufwurf zugewiesen werden. Mit *UCPTdiConfig[x]* erfolgt die Konfiguration der Wippen.

4.1.1 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.

nviButtonDisable

SNVT Typ: SNVT_switch, Index 95

Funktion: Eingangsvariable zum Deaktivieren der Tasten. (Nur für Schalten, Dimmen, Jalousie, Rollläden, Szene)

nviProgDevice

SNVT Typ: SNVT_state, Index 83

Funktion: Eingangsvariable zum Parametrieren des Bedienteils. Nach einer Parametrierung muss *nviProgDevice.bit0* von 0 auf 1 eingestellt werden, damit die neuen Parameter im Bedienteil übernommen werden.

nviProgDevice.bit0 = 0 → *nviProgDevice.bit0* = 1
Übernahme der neuen Parameter.

4.1.2 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.

nvoBCUgroup

SNVT Typ: SNVT_count, Index 8

Funktion: Ausgangsvariable mit dem LON-Interface Typ. Der LON-Interface Typ kann nicht verändert werden.

4.1.3 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: 6135-xx-102, 6129-xx, 6322-xx-101, 6324-xx

6135-xx-102	==>	6135
6129-xx	==>	6129
6322-xx-101	==>	6322
6324-xx	==>	6324

Voreingestellter Wert: 0

UCPTgeneralCP1

UCPT Index: 7, SNVT_state

Funktion: Auswahl des IR-Bereichs der Fernbedienung.

bit0	
0	→ Weiß
1	→ Blau
2	

UCPTlongPressTime

UCPT Index: 71, typedef struct { SNVT_time_sec dimming; SNVT_time_sec sunblind;
SNVT_time_sec scene; SNVT_time_sec universal; }

Funktion: Mit diesem Konfigurationsparameter kann die Zeit in Sekunden für einen langen Tastendruck für Dimmen, Jalousie, Szene und Universell eingegeben werden.
(Voreingestellter Wert: 1.0;2.0;2.0;2.0)

UCPTdiConfig[0]...[4]

UCPT Index: 44, typedef struct (unsigned short Byte[4]) UNVT_str_hex4

Funktion: Diese Konfigurationsparameter bestimmen die Wippenfunktionen und deren Zuordnung zu den Ausgangsvariablen, sowie die Ansteuerlogik der Rückmelde-LEDs. UCPTdiConfig[0] ist fest der Wippe 1, UCPTdiConfig[1] der Wippe 2, UCPTdiConfig[2] der Wippe 3, UCPTdiConfig[3] der Wippe 4 und UCPTdiConfig[4] der Wippe 5 zugeordnet..

UCPTdiConfig[0]	konfiguriert	Wippe 1
UCPTdiConfig[1]	konfiguriert	Wippe 2
UCPTdiConfig[2]	konfiguriert	Wippe 3
UCPTdiConfig[3]	konfiguriert	Wippe 4
UCPTdiConfig[4]	konfiguriert	Wippe 5
UCPTdiConfig[x].Byte[0]	konfiguriert	Funktion der Wippe
UCPTdiConfig[x].Byte[1]	konfiguriert	Funktion der linken Taste
UCPTdiConfig[x].Byte[2]	konfiguriert	Funktion der rechten Taste
UCPTdiConfig[x].Byte[3]	konfiguriert	Funktion der LED

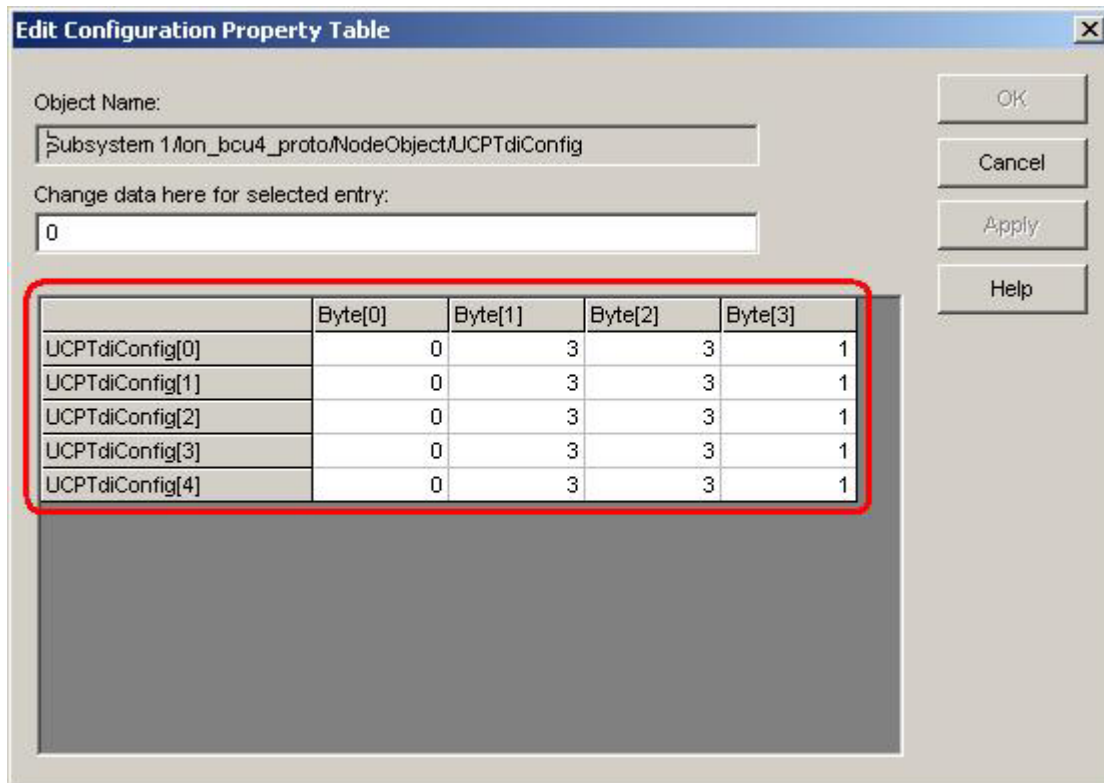


Abbildung 4-1: UCPTdiConfig

UCPTsunblindUP

UCPT Index: 72, SNVT_setting

Funktion: Mit diesem Konfigurationsparameter kann eingestellt werden, welcher SNVT_setting Wert beim Hochfahren der Jalousie / Rollladen gesendet werden soll.
(Voreingestellter Wert: SET_UP 100.0 0.0)

UCPTsunblindDOWN

UCPT Index: 73, SNVT_setting

Funktion: Mit diesem Konfigurationsparameter kann eingestellt werden, welcher SNVT_setting Wert beim Runterfahren der Jalousie / Rollladen gesendet werden soll.
(Voreingestellter Wert: SET_DOWN 100.0 0.0)

UCPTsunblindSTOP

UCPT Index: 74, SNVT_setting

Funktion: Mit diesem Konfigurationsparameter kann eingestellt werden, welcher SNVT_setting Wert beim Anhalten der Jalousie / Rollladen gesendet werden soll.
(Voreingestellter Wert: SET_STOP 0.0 0.0)

UCPTsunblindIDLE

UCPT Index: 75, SNVT_setting

Funktion: Mit diesem Konfigurationsparameter kann eingestellt werden, welcher SNVT_setting Wert für den Ruhezustand der Jalousie / Rollladen gesendet werden soll. *UCPTsunblindIDLE* wird 500ms nach anhalten der Jalousie / Rollladen gesendet, wenn *UCPTsunblindIDLE* ungleich *UCPTsunblindSTOP* ist.
(Voreingestellter Wert: SET_NUL 0.0 0.0)

Ein Wippe ist fest zugeordnet zu einem der Switch Objekte. Jedes Objekt besitzt zwei Ausgangsvariablen (xxx_A; xxx_B). Jeder Taste kann somit ein einzelnes Switch-Objekt zugewiesen werden. Dies ist bei einer toggle Funktion sinnvoll. (siehe Kapitel4.2)

Funktion Beleuchtung Schalten / Tasten: Byte[0] = 0:

Konfiguration der Wippe für Beleuchtung EIN / AUS mit UCPTdiConfig							
Zuordnung Wippen - Funktionsblock		Funktion linke Taste		Funktion rechte Taste		Funktion LED	
Byte[0]	Beschreibung	Byte[1]	Beschreibung	Byte[2]	Beschreibung	Byte[3]	Beschreibung
0	Schalten / Tasten	1	Licht Toggle EIN / AUS (Switch_A_n)	1	Licht Toggle EIN / AUS (Switch_B_n)	1	Linke Wippe rot Rechte Wippe grün
		2	Licht nur EIN (Switch_A_n)	2	Licht nur EIN (Switch_A_n)	2	Linke Wippe grün Rechte Wippe rot
		3	Licht nur EIN (Switch_B_n)	3	Licht nur EIN (Switch_B_n)	3	Linke Wippe rot Rechte Wippe rot
		4	Licht nur AUS (Switch_A_n)	4	Licht nur AUS (Switch_A_n)	4	Linke Wippe grün Rechte Wippe grün
		5	Licht nur AUS (Switch_B_n)	5	Licht nur AUS (Switch_B_n)		
		6	Befehl Automatik (= 0.0 –1) (Switch_A_n)	6	Befehl Automatik (= 0.0 –1) (Switch_A_n)		
		7	Befehl Automatik (= 0.0 –1) (Switch_B_n)	7	Befehl Automatik (= 0.0 –1) (Switch_B_n)		

Beispiele:

A: Die Wippe 1 soll als Schaltfunktion konfiguriert werden. Dabei schaltet die linke Taste das Licht Ein und die rechte Aus. Die LED soll bei einem linken Tastendruck rot und bei einem rechten Tastendruck grün sein.
 UCPTdiConfig[0].Byte[0] = 0 → schalten
 UCPTdiConfig[0].Byte[1] = 2 → Links EIN
 UCPTdiConfig[0].Byte[2] = 4 → Rechts AUS
 UCPTdiConfig[0].Byte[3] = 1 → LED

B: Die Wippe 2 soll als Schaltfunktion konfiguriert werden. Dabei toggelt die linke und die rechte Taste das Licht Ein und Aus. Die LED soll grün leuchten.
 UCPTdiConfig[1].Byte[0] = 0 → schalten
 UCPTdiConfig[1].Byte[1] = 1 → Links toggeln
 UCPTdiConfig[1].Byte[2] = 1 → Rechts toggeln
 UCPTdiConfig[1].Byte[3] = 4 → LED

C: Die Wippe 3 soll als Schaltfunktion konfiguriert werden. Dabei toggelt die linke Taste das Licht Ein und Aus. Die rechte Taste setzt die Beleuchtung in den Automatik-Modus. Die LED soll im toggle-Modus grün und im Automatik-Modus rot leuchten.
 UCPTdiConfig[2].Byte[0] = 0 → schalten
 UCPTdiConfig[2].Byte[1] = 1 → Links toggeln
 UCPTdiConfig[2].Byte[2] = 6 → Rechts Automatik
 UCPTdiConfig[2].Byte[3] = 3 → LED

Funktion Beleuchtung Dimmen: Byte[0] = 1:

Konfiguration der Wippe für Beleuchtung mit Dimmen mit UCPTdiConfig							
Zuordnung Wippen - Funktionsblock		Funktion linke Taste		Funktion rechte Taste		Funktion LED	
Byte[0]	Beschreibung	Byte[1]	Beschreibung	Byte[2]	Beschreibung	Byte[3]	Beschreibung
1	Beleuchtung mit Dimmen	1	Licht Toggle mit Dimmen (Switch_A_n)	1	Licht Toggle mit Dimmen (Switch_B_n)	1	Linke Wippe rot Rechte Wippe grün
			Einschaltwert = Max-Wert		Einschaltwert = Max-Wert		
		2	Licht Toggle mit Dimmen (Switch_A_n)	2	Licht Toggle mit Dimmen (Switch_B_n)	2	Linke Wippe grün Rechte Wippe rot
			Einschaltwert = letzter Ein-Wert		Einschaltwert = letzter Ein-Wert		
		3	Licht nur Heller mit Dimmen (Switch_A_n)	3	Licht nur Heller mit Dimmen (Switch_A_n)	3	Linke Wippe rot Rechte Wippe rot
			Einschaltwert = Max-Wert		Einschaltwert = Max-Wert		
		4	Licht nur Heller mit Dimmen (Switch_B_n)	4	Licht nur Heller mit Dimmen (Switch_B_n)	4	Linke Wippe grün Rechte Wippe grün
			Einschaltwert = Max-Wert		Einschaltwert = Max-Wert		
		5	Licht nur Heller mit Dimmen (Switch_A_n)	5	Licht nur Heller mit Dimmen (Switch_A_n)		
			Einschaltwert = letzter Ein-Wert		Einschaltwert = letzter Ein-Wert		
		6	Licht nur Heller mit Dimmen (Switch_B_n)	6	Licht nur Heller mit Dimmen (Switch_B_n)		
			Einschaltwert = letzter Ein-Wert		Einschaltwert = letzter Ein-Wert		
		7	Licht nur dunkler mit Dimmen (Switch_A_n)	7	Licht nur dunkler mit Dimmen (Switch_A_n)		
		8	Licht nur dunkler mit Dimmen (Switch_B_n)	8	Licht nur dunkler mit Dimmen (Switch_B_n)		
		9	Befehl Automatik (= 0.0 –1) (Switch_A_n)	9	Befehl Automatik (= 0.0 –1) (Switch_A_n)		
		A	Befehl Automatik (= 0.0 –1) (Switch_B_n)	A	Befehl Automatik (= 0.0 –1) (Switch_B_n)		

Beispiele:

A: Die Wippe 1 soll als Dimmfunktion konfiguriert werden. Dabei schaltet die linke Taste das Licht Ein / Heller und die rechte Aus / Dunkler.

UCPTdiConfig[0].Byte[0] = 1 → Dimmen
 UCPTdiConfig[0].Byte[1] = 3 → Links Ein / Heller
 UCPTdiConfig[0].Byte[2] = 7 → Rechts Aus / Dunkler
 UCPTdiConfig[0].Byte[3] = 1 → LED

B: Die Wippe 2 soll als Schaltfunktion konfiguriert werden. Dabei toggelt die linke Taste mit *nvoSwitch_A_n* und die rechte Taste mit *nvoSwitch_B_n* das Licht Ein bzw. Aus. Die LED soll grün leuchten.

UCPTdiConfig[1].Byte[0] = 1 → Dimmen
 UCPTdiConfig[1].Byte[1] = 1 → Links Toggle mit Dimmen
 UCPTdiConfig[1].Byte[2] = 1 → Rechts Toggle mit Dimmen
 UCPTdiConfig[1].Byte[3] = 4 → LED

Funktion Jalousie: Byte[0] = 2:

Konfiguration der Taster für Jalousie mit UCPTdiConfig							
Zuordnung Wippen - Funktionsblock		Funktion linke Taste		Funktion rechte Taste		Funktion LED	
Byte[0]	Beschreibung	Byte[1]	Beschreibung	Byte[2]	Beschreibung	Byte[3]	Beschreibung
2	Jalousie	1	Jalousie AUF (Setting_A_n)	1	Jalousie AUF (Setting_A_n)	1	Linke Wippe rot Rechte Wippe grün
		2	Jalousie AB (Setting_A_n)	2	Jalousie AB (Setting_A_n)	2	Linke Wippe grün Rechte Wippe rot
		3	Jalousie AUF (Setting_B_n)	3	Jalousie AUF (Setting_B_n)	3	Linke Wippe rot Rechte Wippe rot
		4	Jalousie AB (Setting_B_n)	4	Jalousie AB (Setting_B_n)	4	Linke Wippe grün Rechte Wippe grün

Beispiel: Die Wippe 1 soll als Jalousiefunktion konfiguriert werden. Dabei schaltet die linke Taste die Jalousie Auf die rechte Ab. Die LED soll rot leuchten.
 UCPTdiConfig[0].Byte[0] = 2 → Jalousie
 UCPTdiConfig[0].Byte[1] = 1 → Links Auf
 UCPTdiConfig[0].Byte[2] = 2 → Rechts Ab
 UCPTdiConfig[0].Byte[3] = 3 → LED

Funktion Rollladen: Byte[0] = 3:

Konfiguration der Wippe für Jalousie mit UCPTdiConfig							
Zuordnung Wippen - Funktionsblock		Funktion linke Taste		Funktion rechte Taste		Funktion LED	
Byte[0]	Beschreibung	Byte[1]	Beschreibung	Byte[2]	Beschreibung	Byte[3]	Beschreibung
3	Rollladen	1	Rollladen AUF (Setting_A_n)	1	Rollladen AUF (Setting_A_n)	1	Linke Wippe rot Rechte Wippe grün
		2	Rollladen AB (Setting_A_n)	2	Rollladen AB (Setting_A_n)	2	Linke Wippe grün Rechte Wippe rot
		3	Rollladen AUF (Setting_B_n)	3	Rollladen AUF (Setting_B_n)	3	Linke Wippe rot Rechte Wippe rot
		4	Rollladen AB (Setting_B_n)	4	Rollladen AB (Setting_B_n)	4	Linke Wippe grün Rechte Wippe grün

Beispiel: Die Wippe 3 soll als Rollladenfunktion konfiguriert werden. Dabei schaltet die linke Taste den Rollladen Auf die rechte Ab. Die LED soll rot sein.
 UCPTdiConfig[2].Byte[0] = 3 → Rollladen
 UCPTdiConfig[2].Byte[1] = 1 → Links Auf
 UCPTdiConfig[2].Byte[2] = 2 → Rechts Ab
 UCPTdiConfig[2].Byte[3] = 3 → LED

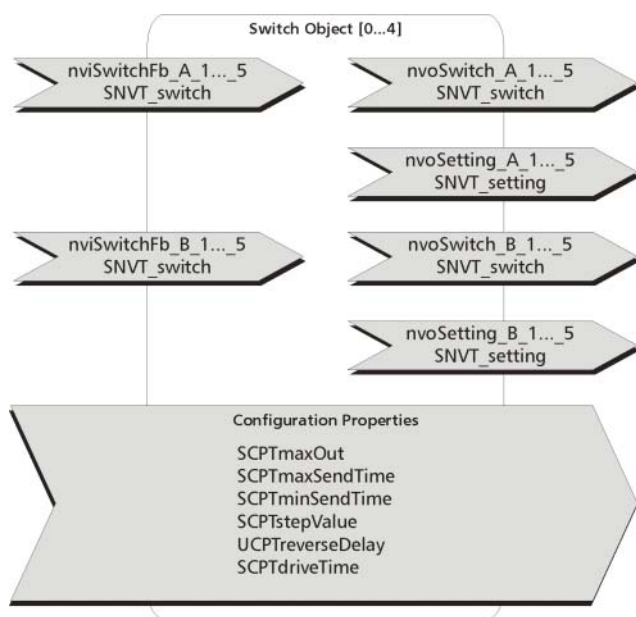
Funktion Szenensteuerung: Byte[0] = 4:

Konfiguration der Taster für Szenensteuerung mit UCPTdiConfig							
Zuordnung Wippen - Funktionsblock		Funktion linke Taste		Funktion rechte Taste		Funktion	
Byte[0]	Beschreibung	Byte[1]	Beschreibung	Byte[2]	Beschreibung	Byte[3]	Beschreibung
4	Szene	01-FE	Szenen-Nummer	01-FE	Szenen-Nummer	1	Linke Wippe rot Rechte Wippe grün
						2	Linke Wippe grün Rechte Wippe rot
						3	Linke Wippe rot Rechte Wippe rot
						4	Linke Wippe grün Rechte Wippe grün

Beispiel: Die Wippe 2 soll als Szenenaufruf konfiguriert werden. Dabei ruft die linke Taste die Szene 5 auf und die rechte Taste die Szene 12. Die LED soll rot leuchten.

UCPTdiConfig[1].Byte[0] = 4 → Szene
 UCPTdiConfig[1].Byte[1] = 5 dez → Szene 5 aufrufen
 UCPTdiConfig[1].Byte[2] = 12 dez → Szene 12 aufrufen
 UCPTdiConfig[1].Byte[3] = 3 → LED

4.2 Switch Object



Fünf identische Switch-Objekte für die Beleuchtungs und Jalousiesteuerung. Die Switch-Objekte werden direkt den Wippen zugeordnet. Dabei gehört zu den einzelnen Objekten:

Switch Object[0] = Wippe 1
 Switch Object[1] = Wippe 2
 Switch Object[2] = Wippe 3
 Switch Object[3] = Wippe 4
 Switch Object[4] = Wippe 5

Die Zuordnung der Wippen zu den einzelnen Funktionen erfolgt wie zuvor beschrieben mit den Konfigurationsparameter UCPTdiConfig[0]...[4] im Node Object. Mögliche Funktionen sind Beleuchtungssteuerung EIN/AUS, Beleuchtungssteuerung EIN/AUS mit Dimmen oder Jalousie AUF/AB mit Automatiklauf.

Jedes Switch Object hat 2 Variablengruppen die durch den Anhang xxx_A und xxx_B unterschieden werden. Dadurch kann jedem Taster eine Netzwerkvariable zugeordnet werden.

4.2.1 Eingangsvariablen Switch Object:

nviSwitchFb_A_1...5, nviSwitchFb_B_1...5

SNVT Typ: SNVT_switch, Index 95

Funktion: Eingangsvariablen für den aktuellen Zustand der mit nvoSwitch_A_1...5 bzw, nvoSwitchFb_B_1...5 angesteuerten Beleuchtungsgruppen.

4.2.2 Ausgangsvariablen Switch Object:

nvoSwitch_A_1...5, nvoSetting_A_1...5, nvoSwitch_B_1...5, nvoSetting_B_1...5

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. Jeder Wippe kann ein Switch Object zugewiesen werden. Jedes Switch Object hat zwei verschiedene Variablengruppen, welche sich durch den Anhang *_A* bzw. *_B* unterscheiden. Dadurch kann jedem Taster eine Netzwerkvariable zugeordnet werden.

Mit

UCPTdiConfig[x].Byte[0] werden Wippen einer Funktion zugeordnet.

UCPTdiConfig[x].Byte[1] weist der linken Taste eine Funktion zu.

UCPTdiConfig[x].Byte[2] weist der rechten Taste eine Funktion zu.

UCPTdiConfig[x].Byte[3] weist der LED die Funktion zu.

Schalten/ Tasten

UCPTdiConfig[x].Byte[0] = 00_{hex}

Taste gedrückt/ nicht gedrückt

UCPTdiConfig[x].Byte[1...2] = 01_{hex}

Taste gedrückt	nvoSwitch_A_x.value	= SCPTmaxOut
	nvoSwitch_A_x.state	= 1
	nvoSetting_A_x.function	= SET_ON;
	nvoSetting_A_x.setting	= SCPTmaxOut;

Taste nicht gedrückt	nvoSwitch_A_x.value	= 0
	nvoSwitch_A_x.state	= 0
	nvoSetting_A_x.function	= SET_OFF;
	nvoSetting_A_x.setting	= 0;

UCPTdiConfig[x].Byte[1...2] = 02_{hex}

Taste gedrückt	nvoSwitch_B_x.value	= SCPTmaxOut
	nvoSwitch_B_x.state	= 1
	nvoSetting_B_x.function	= SET_ON;
	nvoSetting_B_x.setting	= SCPTmaxOut;

Taste nicht gedrückt	nvoSwitch_B_x.value	= 0
	nvoSwitch_B_x.state	= 0
	nvoSetting_B_x.function	= SET_OFF;
	nvoSetting_B_x.setting	= 0;

Beleuchtung Toggle

UCPTdiConfig[x].Byte[1] = 03_{hex}

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

Beleuchtung EIN	nvoSwitch_A_x.value	= SCPTmaxOut
	nvoSwitch_A_x.state	= 1
	nvoSetting_A_x.function	= SET_ON;
	nvoSetting_A_x.setting	= SCPTmaxOut;

Beleuchtung AUS	nvoSwitch_A_x.value	= 0
	nvoSwitch_A_x.state	= 0
	nvoSetting_A_x.function	= SET_OFF;
	nvoSetting_A_x.setting	= 0;

Softwarebeschreibung

UCPTdiConfig[x].Byte[2] = 03_{hex}

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

Beleuchtung EIN	nvoSwitch_B_x.value	= SCPTmaxOut
	nvoSwitch_B_x.state	= 1
	nvoSetting_B_x.function	= SET_ON;
	nvoSetting_B_x.setting	= SCPTmaxOut;
Beleuchtung AUS	nvoSwitch_B_x.value	= 0
	nvoSwitch_B_x.state	= 0
	nvoSetting_B_x.function	= SET_OFF;
	nvoSetting_B_x.setting	= 0;

Beleuchtung EIN

UCPTdiConfig[x].Byte[1...2] = 04_{hex}

Jede Tastbetätigung führt zum Einschalten der Beleuchtung

Beleuchtung EIN	nvoSwitch_A_x.value	= SCPTmaxOut
	nvoSwitch_A_x.state	= 1
	nvoSetting_A_x.function	= SET_ON;
	nvoSetting_A_x.setting	= SCPTmaxOut;

UCPTdiConfig[x].Byte[1...2] = 05_{hex}

Jede Tastbetätigung führt zum Einschalten der Beleuchtung

Beleuchtung EIN	nvoSwitch_B_x.value	= SCPTmaxOut
	nvoSwitch_B_x.state	= 1
	nvoSetting_B_x.function	= SET_ON;
	nvoSetting_B_x.setting	= SCPTmaxOut;

Beleuchtung AUS

UCPTdiConfig[x].Byte[1...2] = 06_{hex}

Jede Tastbetätigung führt zum Ausschalten der Beleuchtung

Beleuchtung AUS	nvoSwitch_A_x.value	= 0
	nvoSwitch_A_x.state	= 0
	nvoSetting_A_x.function	= SET_OFF;
	nvoSetting_A_x.setting	= 0;

UCPTdiConfig[x].Byte[1...2] = 07_{hex}

Jede Tastbetätigung führt zum Ausschalten der Beleuchtung

Beleuchtung AUS	nvoSwitch_B_x.value	= 0
	nvoSwitch_B_x.state	= 0
	nvoSetting_B_x.function	= SET_OFF;
	nvoSetting_B_x.setting	= 0;

Automatik

UCPTdiConfig[x].Byte[1...2] = 08_{hex}

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

UCPTdiConfig[x].Byte[1...2] = 08_{hex}

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

Dimmen

UCPTdiConfig[x].Byte[0] = 01_{hex}

Beleuchtung Togglen 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

Softwarebeschreibung

Beleuchtungszustand wird der .value-Wert der Switch-Variablen in Prozentschritten von UCPTstepValue erhöht oder verringert und zwar so lange wie die Taste gedrückt wird.

UCPTdiConfig[x].Byte[1] = 01_{hex}

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

Die Variablen nvoSetting_A_x werden in dieser Einstellung nicht verändert.

UCPTdiConfig[x].Byte[2] = 01_{hex}

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

Die Variablen nvoSetting_B_x werden in dieser Einstellung nicht verändert.

Beleuchtung Togglen mit Dimmen, Einschaltwert = letzter Ein-Wert

UCPTdiConfig[x].Byte[1] = 02_{hex}

Funktion wie bei 01_{hex}, nur mit dem Unterschied, dass beim Einschalten der Beleuchtung nicht der Wert SCPTmaxOut, sondern der letzte Einschaltwert angenommen wird. Der kleinste Einschaltwert ist hierbei auf 20% begrenzt.

Beleuchtung EIN mit Heller-Dimmen, Einschaltwert = max. Wert

Ist die Beleuchtung ausgeschaltet, dann führt eine Tastbetätigung zum sofortigen Einschalten der Beleuchtung. 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.

UCPTdiConfig[x].Byte[1...2] = 03_{hex}

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

UCPTdiConfig[x].Byte[1...2] = 04_{hex}

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

Beleuchtung EIN mit Heller-Dimmen, Einschaltwert = letzter Ein-Wert

UCPTdiConfig[x].Byte[1...2] = 05_{hex}, 06_{hex}

Funktion wie bei 03_{hex}, 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.

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.

UCPTdiConfig[x].Byte[1...2] = 07_{hex}

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

UCPTdiConfig[x].Byte[1...2] = 08_{hex}

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

Automatik

UCPTdiConfig[x].Byte[1..2] = 09_{hex}

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

UCPTdiConfig[x].Byte[1...2] = 0A_{hex}

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

Jalousie

UCPTdiConfig[x].Byte[0] = 02_{hex}

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.

UCPTdiConfig[x].Byte[1,2] = 01_{hex}

Jalousie öffnen	nvoSetting_A_x.function	= UCPTsunblindUP;
Jalousie anhalten	nvoSetting_A_x.function	= UCPTsunblindSTOP;

UCPTdiConfig[x].Byte[1,2] = 03_{hex}

Jalousie öffnen	nvoSetting_B_x.function	= UCPTsunblindUP;
Jalousie anhalten	nvoSetting_B_x.function	= UCPTsunblindSTOP;

Mit einer Verzögerung von 500ms kann nach senden des Befehls UCPTsunblindSTOP der Befehl UCPTsunblindIDLE gesendet, wenn dieser ungleich UCPTsunblindSTOP ist.

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.

UCPTdiConfig[x].Byte[1,2] = 02_{hex}

Jalousie schließen	nvoSetting_A_x.function	= UCPTsunblindDOWN;
Jalousie anhalten	nvoSetting_A_x.function	= UCPTsunblindSTOP;

UCPTdiConfig[x].Byte[1,2] = 04_{hex}

Jalousie schließen	nvoSetting_B_x.function	= UCPTsunblindDOWN;
Jalousie anhalten	nvoSetting_B_x.function	= UCPTsunblindSTOP;

Mit einer Verzögerung von 500ms kann nach senden des Befehls UCPTsunblindSTOP der Befehl UCPTsunblindIDLE gesendet, wenn dieser ungleich UCPTsunblindSTOP ist.

Rollladen

UCPTdiConfig[x].Byte[0] = 03_{hex}

Rollladen AUF

Im Konfigurationsmodus „Rollladen AUF“ werden nur die nvoSetting-Variablen verändert und gesendet. Kurze Tastbetätigungen (< 2 s) startet den Automatiklauf und steuert den Rollladen für die Zeit SCPTdriveTime dauerhaft in Richtung öffnen an. Der Automatiklauf kann durch erneute Tastbetätigung angehalten werden. Mit einer langen Tastbetätigung (> 2 s) kann die Position des Rollladens individuell eingestellt werden.

UCPTdiConfig[x].Byte[1,2] = 01_{hex}

Rollladen öffnen	nvoSetting_A_x.function	= UCPTsunblindUP;
Rollladen anhalten	nvoSetting_A_x.function	= UCPTsunblindSTOP;

UCPTdiConfig[x].Byte[1,2] = 03_{hex}

Rollladen öffnen	nvoSetting_B_x.function	= UCPTsunblindUP;
Rollladen anhalten	nvoSetting_B_x.function	= UCPTsunblindSTOP;

Mit einer Verzögerung von 500ms kann nach senden des Befehls UCPTsunblindSTOP der Befehl UCPTsunblindIDLE gesendet, wenn dieser ungleich UCPTsunblindSTOP ist.

Rollladen AB

Im Konfigurationsmodus „Rollladen AB“ werden nur die nvoSetting-Variablen verändert und gesendet. Kurze Tastbetätigungen (< 2 s) startet den Automatiklauf und steuert den Rollladen für die Zeit SCPTdriveTime dauerhaft in Richtung schließen an. Der Automatiklauf kann durch erneute Tastbetätigung angehalten werden. Mit einer langen Tastbetätigung (> 2 s) kann die Position des Rollladens individuell eingestellt werden.

UCPTdiConfig[x].Byte[1,2] = 02_{hex}

Rollladen schließen	nvoSetting_A_x.function	= UCPTsunblindDOWN;
Rollladen anhalten	nvoSetting_A_x.function	= UCPTsunblindSTOP;

UCPTdiConfig[x].Byte[1,2] = 04_{hex}

Rollladen schließen	nvoSetting_B_x.function	= UCPTsunblindDOWN;
Rollladen anhalten	nvoSetting_B_x.function	= UCPTsunblindSTOP;

Mit einer Verzögerung von 500ms kann nach senden des Befehls UCPTsunblindSTOP der Befehl UCPTsunblindIDLE gesendet, wenn dieser ungleich UCPTsunblindSTOP ist.

Softwarebeschreibung

4.2.3 Konfigurationsparameter Switch Object:

SCPTmaxOut

SCPT Index: 93, SNVT_lev_cont

Funktion: Dieser Konfigurationsparameter bestimmt den maximalen Ausgabewert der Variablen nvoSwitch.value. (Voreingestellter Wert: 100.0)

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. (Voreingestellter Wert: 5.0)

UCPTreverseDelay

UCPT Index: 14, SNVT_count

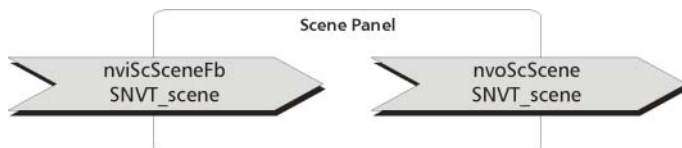
Funktion: Dieser Konfigurationsparameter definiert die Umschaltverzögerung bei Drehrichtungs-umkehr der Jalousiemotoren in ms. Damit wird ein Wechselbefehl z.B. von nvoSetting = SET_UP auf nvoSetting = SET_DOWN verzögert ausgegeben. (Voreingestellter Wert: 500 ms)

SCPTdriveTime

UCPT Index: 45, SNVT_time_sec

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

4.3 Scene Panel



4.3.1 Eingangsvariablen Scene Panel

nviScSceneFb

SNVT Typ: SNVT_scene, Index 115

Funktion: Eingangsvariable mit der aktuellen Beleuchtungsszene im Raum.

4.3.2 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[x].Byte[0] = 3 werden Taster dem Scene-Object zugeordnet.

UCPTdiConfig[x].Byte[1...2] = 01_{hex} - FE_{hex} weist diesen Tastern dann eine Szenennummer zu, die bei kurzen Tastbetätigungen mit SC_RECALL aufgerufen wird. Bei langen Tastbetätigungen (> 2s) wird die Szene mit SC_LEARN neu gelernt. Es können auch mehrere Tasten dem Scene Panel zugeordnet werden.