Functional Profile

SOMFY animeo®
LON RTS MODULE 433 MHZ WM
Ref. 1860239

SOMFY animeo®
LON RTS MODULE 447 MHZ WM
Ref. 1860240

Version 1.02
Revision date: 09.06.2015
Ref.Nr.: 5118943A

Somfy SAS
50 avenue du Nouveau Monde
74300 Cluses
France

Table of Content

1. Document History

version	Modification	reason	validity date
1.0	First Version		03.12.2014
1.01	Appendix added		09.01.2015
1.02	Device "LON RTS Module 433MHz" added		12.05.2015

2. Hardware

The device has three LEDs and one button. The latter and the yellow LED are used for service purpose as defined by LONMARK.

The green LED signals the operation of the power supply.

The orange LED signals the state of the radio receiver. Blinking means that no transmitter is registered. As soon as at least one radio transmitter is known the LED stops blinking and signals further on the reception of a valid radio frame by an one second on wink

3. Node Object #0000

`UFPTnodeObject #0000`

3.1. Overview

The SOMFY Node Object inherits all mandatory and some optional members from the LonMark® Node Object # 0000. Various SOMFY specific members have been added.

The Node Object functional profile describes a special type of functional block—called the *Node Object functional block*—that is used by network tools to test and manage all the functional blocks on a device.

The Node Object functional block may also be used to initiate the collecting of remote control addresses. Any remote control can be linked only once to one interface device. Although possible it is not advisable to link a radio control twice to a LON system. If required this functionality should be accomplished in LON.

The Node Object functional block includes a mandatory **nviRequest** input network variable and a mandatory **nvoStatus** output network variable. Other devices and applications may request a Node Object function by sending a request to the **nviRequest** network variable. Upon receiving an update to the **nviRequest** network variable, the request is processed and the **nvoStatus** network variable is updated with either the results of the request, an in-process indication, or an error indication. The definition of the **nviRequest** network variable includes an object ID field to allow the Node Object to report status and alarm conditions for all functional blocks on a device.

3.2. Functional-Block Details

```
                    ┌─────────────────────────────┐
                    │        Node Object          │
                    ├─────────────────────────────┤
                    │      Network Variables       │
        ┌────────────────┐
        │  nviRequest     \
        │  SNVT_obj_request /
        └────────────────┘        ┌──────────────────┐
                                  │  nvoStatus        \
                                  │  SNVT_obj_status  /
        ┌────────────────┐        └──────────────────┘
        │  nviProgMode    \
        │  SNVT_switch    /
        └────────────────┘        ┌──────────────────┐
                                  │  nvoFileDirectory \
                                  │  SNVT_address     /
                    ├─────────────────────────────┤
                    │  Configuration Properties   │
        ┌─────────────────────────────────────┐
        \              nciLocation             \
        /                                       /
        \             nciDevMajVer              \
        /                                       /
        \             nciDevMinVer              \
        /                                       /
                    └─────────────────────────────┘
```
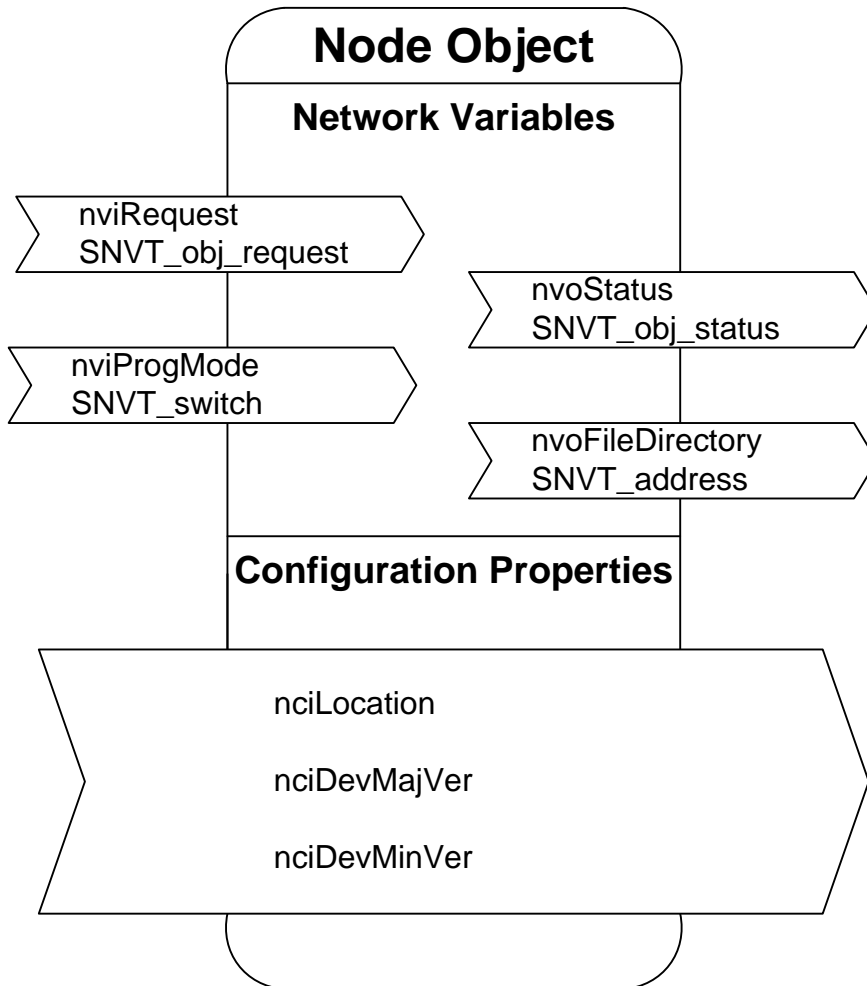
Table 1 SNVT Details

NV (M/O)*	Variable Name	SNVT/UNVT Name	SNVT/UNVT Index	Description
1 (M)	nviRequest	SNVT_obj_request	92	Requests a particular mode for a particular functional block in the device
2 (M)	nvoStatus	SNVT_obj_status	93	Reports the status of the requested functional block in the device
8 (M)	nvoFileDirectory	SNVT_address	114	Address for the file directory containing descriptors for configuration files
#1 (O)	nviProgMode	SNVT_switch	95	Activation of programming mode to collect further radio transmitter devices

* M = mandatory, O = optional

Table 2 SCPT Details

(M/O)*	SCPT/UCPT Name NV Name Type or SNVT	SCPT/ UCPT Index	Associated NVs **	Description
O	SCPTlocation nciLocation SNVT_str_asc (36)	17	Entire Functional Block	Provides descriptive physical location information related to the entire device.
O	SCPTdevMajVer nciDevMajVer unsigned short	165	Entire Functional Block	The major version number for the device
O	SCPTdevMinVer nciDevMinVer unsigned short	166	Entire Functional Block	The minor version number for the device

*   M = mandatory, O = optional
** List of NVs to which this configuration property applies.
    An "(M)" means that the CP is mandatory if the NV (to which it applies) is implemented.
    An "(O)" means that the CP is optional if the NV (to which it applies) is implemented.

3.3. Network Variables

3.3.1. Request Input

```
network input SNVT_obj_request nviRequest;
```

This input network variable provides the mechanism to request an operation or a mode for a functional block within a device. For a listing of all possible request codes, and for the meaning of the function codes for **SNVT_obj_request**, see the SNVT Master List.

A request consists of an object ID (the **object_id** field) and an object request (the **object_request** field). The object ID is the functional block index for a functional block on the device. If a device has a Node Object functional block, its functional block index must be zero. The remaining functional blocks are numbered sequentially, starting with one.

The object request specifies a request function for the functional block identified by the object ID. The **object_request_t** definition in the SNVT Master List defines the available request functions; the following requests are the only mandatory request functions:

> **RQ_NORMAL**
>
> **RQ_UPDATE_STATUS**
>
> **RQ_REPORT_MASK**

If an **nviRequest** update specifies an unsupported request function, the **nvoStatus** output network variable must be updated with the **invalid request** field set to one. Support for the object-disable, self-test, override, and alarm reporting request functions is not required.

The request functions are defined as follows:

RQ_NORMAL If the specified functional block was in the disabled or overridden state, this request cancels that state and returns the functional block to normal operation. If the functional block was already in the normal state, a request to enter the normal state is not an error. After device reset, the state of functional blocks on the device is application-specific. An **RQ_NORMAL** request that specifies the Node Object functional block index is a request for all functional blocks in the device to leave the disabled and overridden states.

RQ_UPDATE_STATUS Requests the status of the specified functional block to be sent to the **nvoStatus** output network variable. The state of the functional block is unchanged. A **RQ_ UPDATE_STATUS** request that specifies the Node Object functional block is a request for the status of the device and all functional blocks on the device. The status bits of the Node Object (with the exception of **invalid_request** and **invalid_id**) are defined to be the inclusive-OR of the status bits of all the other functional blocks in the device; with the possible addition of error conditions and other conditions attributed to the device as a whole, rather than to any individual functional block. For example, if **comm_failure** is supported for the Node Object, then it should be set when reporting the Node Object functional block status whenever any of the functional blocks in the device reports communications failure, as well as when there is a communications failure at the device level.

RQ_REPORT_MASK Requests a status mask reporting the status bits that are supported by the specified functional block to be sent to the **nvoStatus** output network variable. A one bit in the status mask means that the device may set the corresponding bit in the object status when the condition defined for that bit occurs. A zero bit in the status mask means that the bit is never set by the device. For example, if object disable (**RQ_DISABLED**) is not supported for a functional block, the **disabled** bit in the status mask must be zero for that functional block. If self-test (**RQ_SELF_TEST**) is not supported for a functional block, the **fail_self_test** and **self_test_in_progress** bits in the status mask must be zero for that functional block. If alarm reporting (**RQ_UPDATE_ALARM** or asynchronous notification) is not supported, the **in_alarm** bit in the status mask must be zero for that functional block. A **RQ_REPORT_MASK** request that specifies the Node Object functional block requests a status mask that is the inclusive-OR of supported status bits for the device and all functional blocks on the device.

Valid Range

The valid range is any value within the defined limits of **SNVT_obj_request.**

Default Value

The default value is undefined.

Configuration Considerations

None specified.

3.3.2. **Status Output**

```
network output SNVT_obj_status nvoStatus;
```

This output network variable reports the status for any functional block on a device. It is also used to report the status of the entire device and all functional blocks on the device.

A status update consists of an object ID (the **object_id** field) and multiple status fields. The object ID is the functional block index as described under **nviRequest**. If the object ID is zero, the status of the device itself and all functional blocks on the device are reported.

The status fields are one-bit bitfields. The only required status fields are the **report_mask**, **invalid_id** and **invalid_request** fields; all other status fields are optional. If an error condition is active for a reported functional block, the **out_of_limits** field is set to one. Following is a description of the required status fields. See the *SNVT Master List* for a description of the optional fields.

invalid_request Set to one if an unsupported request code (**RQ_***xxx*) is received on the **nviRequest** input network variable.

invalid_id Set to one if a request is received for a functional block index that is not defined in the device. No further checking of the request code is required when set to one.

report_mask Set to one if a **RQ_REPORT_MASK** request is received by the **nviRequest** input network variable, and the **nvoStatus** output network variable is set to contain the status mask. The *status mask* is a **nvoStatus** value that describes the status bits that are supported beyond the three mandatory status bits. The status mask consists of all fields in the **nvoStatus** output network variable, with the exception of the **report_mask**, **invalid_id** and **invalid_request** fields. A one bit in the mask means that the functional block may set the corresponding bit in the **nvoStatus** output network variable when the condition defined for that bit occurs. A zero bit means that the functional block may never set the bit.

Valid Range

The valid range is any value within the defined limits of **SNVT_obj_status** with the exception that the **report_mask**, **invalid_id** and **invalid_request** fields must be set to one.

Default Value

The default value must be the actual status of the device for all supported fields. All other fields must be set to zero. The application must update the status such that a polling of the status, following the request, returns a reasonable value.

Configuration Considerations

None.

When Transmitted

The output variable is transmitted when either of the following conditions occurs:

A request is received by the **nviRequest** input network variable.

Default Service Type

The default service type is acknowledged.

3.3.3. **File Directory**

```
network output SNVT_address nvoFileDirectory;
```

This output network variable reports the starting address of the configuration-file directory on a Neuron hosted device. It is used when configuration properties are implemented within configuration files accessed by ANSI/EIA/CEA-709.1 Read Memory and Write Memory network-management messages. If an **nvoFileDirectory** output network variable is implemented on a device, all files on the device must be accessible using network management read/write messages. For more details, see *Configuration Properties* within the *LONMARK Application-Layer Interoperability Guidelines*.

This output network variable must be implemented in the Node Object functional block if the device supports the LONWORKS FTP with random and sequential access method. It must not be implemented if the device supports the LONWORKS FTP with sequential access or the direct memory read/write access methods for data files.

Valid Range

The valid range for the file directory address is any value within the user-data memory space of a Neuron Chip or Smart Transceiver.

Default Value

The typical default value is FS_NUL.

Configuration Considerations

The Node Object implements the file-request and file-position network variables as inputs, and the file-status network variable as an output. The device can therefore act as the Sender or the Receiver in a file transfer, but it cannot act as the Initiator of a file transfer using these network variables.

When Transmitted

The output variable is transmitted when either of the following conditions occurs:

☐ During file transfer

☐ When polled

Default Service Type

The default service type is unspecified. Network tools may wish to poll this network variable for values.

3.3.4. **Programming Mode**

```
network input SNVT_switch nviProgMode;
```

This input network variable activates the programming mode to collect further radio transmitter devices. Once switched on the device listens for any programming telegram in the air. As soon as a **new** radio transmitter message was received the NV changes back to off and stores the address of the transmitter in the next free memory space.

It is not possible to link one radio transmitter multiple times to one interface device.

Valid Range

The valid range is the range of **SNVT_switch**.

Default Value

The default value is "0.0 0" (value=0; state=0).

Configuration Considerations

Don't forget to save the received address (es)! After a timeout of one hour the device aborts the programming. See nciTransmitterAddr in RadioCtrl Functional Block(s).

3.4. Configuration Properties

3.4.1. Location Label (optional)

```
SCPTlocation cp_family nciLocation;
```

This configuration property identifies the subsystem containing the device. It provides a more detailed description of the device location than can be provided by the Neuron Chip's 6-byte location string. This allows network-recovery tools to recover subsystem information from a device. The subsystem may be a simple location name, or may be a hierarchical subsystem name. If a hierarchical subsystem name is specified, the subsystem hierarchy components must be separated by periods (**.**). For example, a device may have an **nciLocation** value of "Bldg 1**.**Floor 2**.**Rm 29", representing the Bldg 1/Floor 2/Rm 29 subsystem. Periods must not otherwise be used in the **nciLocation** value. Other characters that cannot be used in the **nciLocation** value are the backslash (\ ), colon (**:**), forward slash (**/**), or double-quotation (**"**) characters (however, stylistic quotation characters can be used, such as „ , " , " , « , and »). For very large networks, subsystem numbers may be used instead of subsystem names, for example: "1**.**2**.**29". This allows deeply nested hierarchies to fit within the 31-character limit for **SCPTlocation**.

If a device has multiple locations, such as a device with multiple remote sensors, each of the functional blocks on the device may also implement a **SCPTlocation** configuration property to identify the location of each of the remote components. The **SCPTlocation** configuration property associated with the Node Object identifies the location of the device itself, whereas the other **SCPTlocation** configuration properties identify the locations of their respective hardware components.

If a device does not have a Node Object functional block, a **SCPTlocation** configuration property that applies to the device may be implemented to document the device location.

Valid Range

Any NUL-terminated ASCII string up to 31 bytes of total length (including NUL). The string must be truncated if the length does not allow the thirty-first character to be the NUL (0).

Configuration Requirements/Restrictions

This CP has no modification restrictions. It can be modified at any time.

3.4.2. Device Major Version

```
network input config SCPTdevMajVer nciDevMajVer;
```

This configuration property provides the major version number of a device.

The major version number is incremented when the network interface for the device changes.

Valid Range

Any integer number from 0 to 255.

Default Value

The default value is zero.

Configuration Requirements/Restrictions

Read only.

3.4.3. **Device Minor Version**

```
network input config SCPTdevMinVer nciDevMinVer;
```

This configuration property provides the minor version number of a device.

The minor version number is incremented when the network interface remains the same, but the device has a different behaviour.

Valid Range

Any integer number from 0 to 255.

Default Value

The default value is zero.

Configuration Requirements/Restrictions

Read only. Only online available.

4. Radio Control #3200

4.1. Overview

The SOMFY Radio Control inherits all mandatory and some optional members from the LonMark® Switch Object # 3200. Various SOMFY specific members have been added.

This document describes the profile for a Radio Control object. The profile is used preferably for sunblind radio controls composed of switch-up and switch-down hardware. The Radio Control object can be used for both closed and open loop applications.

The Functional Block "Radio Control" exists 30 times in the animeo LON RTS Module Each could be set individually and separately from the others and can be linked to exactly one remote control. On one interface device any remote control can be linked only once.

4.2. Functional-Block Details

Radio Control

Network Variables

nvoSwitch
SNVT_switch

nviSwitchFB
SNVT_switch

nvoSetting
SNVT_setting

nviLearnAddr
UNVT_transmitter_addr

Configuration Properties

nciLocation

nciTransmitterAddr

nciLongOperation

nciSettingUp

nciSettingMy

nciSettingDown

nciStepAngle

nciScrollStep
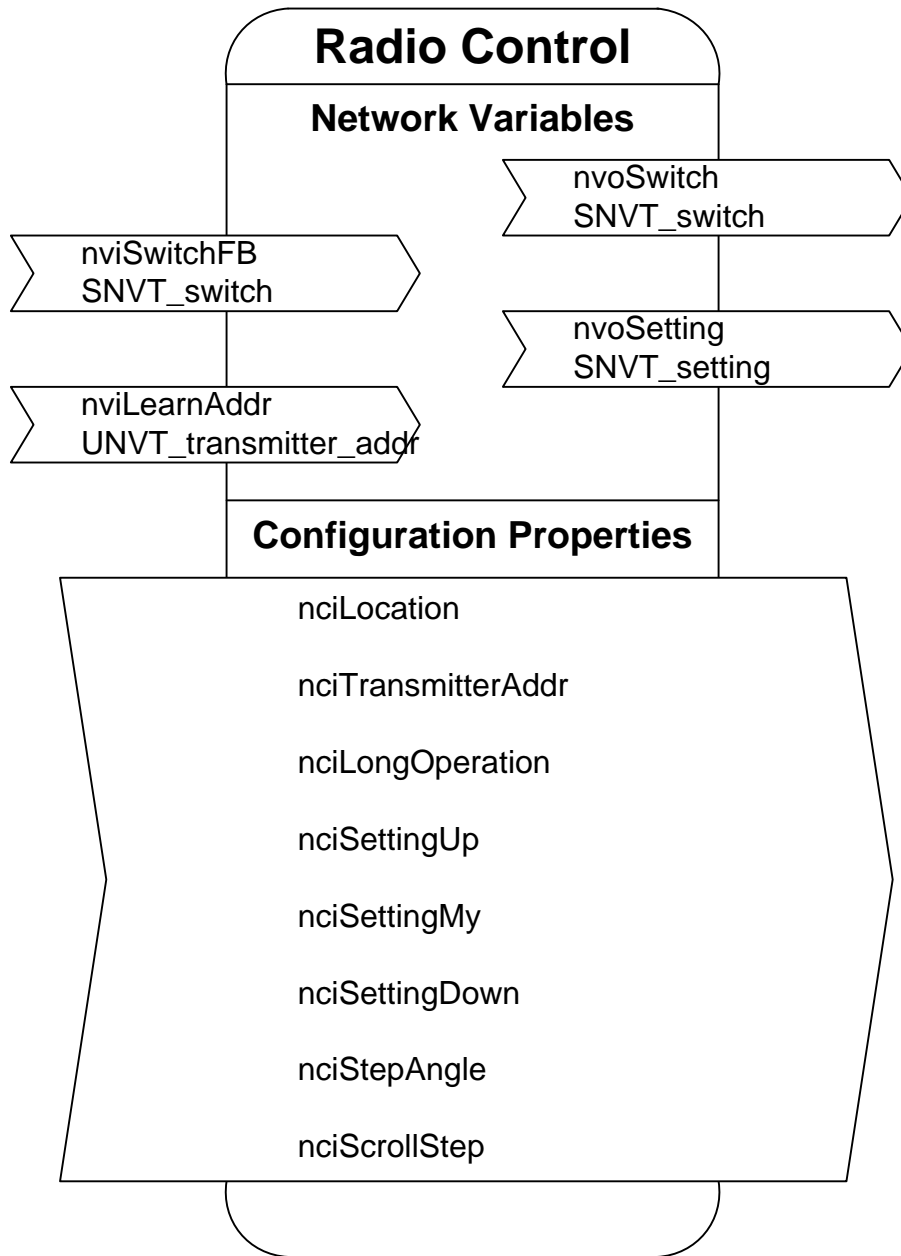
Table 1 SNVT Details

NV (M/O)*	Variable Name	SNVT Name	SNVT Index	Description
1 (M)	nvoSwitch	SNVT_switch	95	Switch output value linked to Up/Down/My radio buttons
2 (O)	nviSwitchFb	SNVT_switch	95	Switch feedback input
3 (O)	nvoSetting	SNVT_setting	117	Setting output
# 1(O)	nviLearnAddr	UNVT_transmitter_addr	1	Address of registered radio transmitter device

* M = mandatory, O = optional

Table 2 SCPT Details

Man. Opt. *	SCPT Name NV Name Type or SNVT	SCPTIndex	Associated NVs	Description
(O)	SCPTlocation nciLocation SNVT_str_asc (36)	1	Entire Functional Block	Provides descriptive physical location information related to the object.
(O)	UCPTtransmitterAddr nciTransmitterAddr UNVT_transmitter_addr	#1	#nv1 (O)	Address of registered radio transmitter device
(O)	UCPTdlyStartLong nciLongOperation SNVT_time_sec (107)	#2	Entire Functional Block	Time after which long operation command is generated
(O)	UCPTsettingUp nciSettingUp SNVT_setting (117)	#3	nv3 (O)	Values which are sent via nvoSetting if Up command was received
(O)	UCPTs ettingMy nciS ettingMy SNVT_setting (117)	#4	nv3 (O)	Values which are sent via nvoSetting if My command was received
(O)	UCPTsettingDown nciSettingDown SNVT_setting (117)	#5	nv3 (O)	Values which are sent via nvoSetting if Down command was received
(O)	UCPTstepAngle nciStepAngle SNVT_angle_deg (104)	#6	nv3 (O)	Rotation distance for one tilting step
(O)	UCPTscrollStep nciScrollStep SNVT_level_cont (21)	#7	nv3 (O)	Percentage of value change for tilting and dimming

* M = mandatory, O = optional

It should be mandatory for CPs that mandatory for an NV that is also mandatory. This is also valuable for CPs that applies to the Entire Functional Block.

** List of NVs to which this configuration property applies.

An "(M)" means that the CP is mandatory if the NV (to which it applies) is implemented.

An "(O)" means that the CP is optional if the NV (to which it applies) is implemented.

4.3. Network Variables

4.3.1. Switch Output

```
network output SNVT_switch nvoSwitch;
```

This output network variables provides the switch output linked to the radio buttons. The Up-button switches on, the Down-button switches off and the My-button toggles the output.

Depending on the used radio transmitter the value can be increased/decreased for dimming purpose with the scroll wheel or by long operation of buttons.

Valid Range

The valid range is the range of SNVT_switch. State 0 means OFF, 1 means ON. The range of the 8-bit intensity value is 0 - 200 (0 - 100% in 0,5% steps). Minimum level is value 0.

When Transmitted

Whenever the hardware received a corresponding radio message.

Update Rate

There is no maximum update rate. The default minimum update rate is 0 ms. Minimum update rate is configurable.

Default Service Type

4.3.2. Switch Feedback Input

```
network input SNVT_switch nviSwitchFb;
```

This input network variable provides the feedback from other devices. This is important for toggling. It does not initiate any action.

Valid Range

Valid range is the range of SNVT_switch.

Default Value

The default value is state = -1, value = 0,0.

4.3.3. **Setting Output**

```
network output SNVT_setting nvoSetting;
```

When blinds are controlled by a controller, such as a sunblind controller, the setting output is used to change the mode and/or the setpoint of the controller. The controller can be turned ON or OFF and the setpoint can be adjusted.

Valid Range

The Valid Range is given by the interpretation of the "SNVT_setting related to sunblinds" as shown in the tables it the chapter "Additional Considerations" in the Sunblind Actuator Profile.

When Transmitted

Whenever the hardware received a corresponding radio message.

Update Rate

There is no maximum update rate. The default minimum update rate is 0 ms. Minimum update rate is configurable.

Default Service Type

4.3.4. **Learn Address**

```
network input UNVT_transmitter_addr nviLearnAddr_n;
```

This input network variable shows the address of the radio transmitter linked actually to this Functional Block.
It can be used as well to learn a (new) transmitter address if the actual one has been deleted before by entering Zero. The device listens then for any programming telegram in the air. As soon as a **new** radio transmitter message has been received the NV displays it and stores the address of the transmitter.

It is not possible to link one radio transmitter multiple times to one interface device.

Valid Range

0x000001 ... 0xFFFFFF

Default Value

0xFF000000 (raw value)

Configuration Considerations

Don't forget to save the received address (es)!)! After a timeout of one hour the device aborts the learning. See nciTransmitterAddr.

To learn in the address of a Somfy RTS device nviLearnAddr_n has to be set to 0.

4.4.   Configuration Properties

4.4.1. **Location Label**

```
network input config SNVT_str_asc nciLocation;
```

This input configuration network variable is used to store ASCII text. It provides more space for descriptive location information.

Valid Range

Any NUL terminated ASCII string of 31 bytes total length.

Default Value

An ASCII string: "Radio Transmitter n".

SCPT Reference

SCPTlocation #17

4.4.2. **Transmitter Address**

```
network input config UCPTtransmitterAddr
nciTransmitterAddr
```

Address of registered radio transmitter device.

Valid Range

0x000001 ... 0xFFFFFF

Default Value

0xFF000000 (raw value)

Configuration Requirements/Restrictions

Values above the valid range and duplicates in one device are rejected. For verification compare to nviLearnAddr.

If a transmitter address is learned (see above nviLearnAddr) this value should be read from device in order to store its value in the LNS data base.

SNVT Reference

Unsigned quad

4.4.3. **Long Operation**

```
network input config UCPTdlyStartLong nciDlyStartLong
```

A push button activation for longer than the adjusted time outputs the setting specified in the corresponding configuration property.

A shorter activation outputs a corresponding up/down angle step.

Valid Range

The valid range of SNVT_time_sec.

Default Value

0,3 seconds.

Configuration Requirements/Restrictions

RTS frames are typically repeated every 140 ms, thus only multiples of these times are considered precisely.

SNVT Reference

SNVT_time_sec #107

4.4.4. **Setting Down**

```
network input config UCPTsettingDown nciSettingDown
```

Values which are sent via nvoSetting if down command has been received.

Valid Range

The valid range of SNVT_setting.

Default Value

SET_DOWN, INVALID, INVALID.

SNVT Reference

SNVT_setting #117

4.4.5. **Setting Up**

```
network input config UCPTsettingUp nciSettingUp
```

Values which are sent via nvoSetting if up command has been received.

Valid Range

The valid range of SNVT_setting.

Default Value

SET_UP, INVALID, INVALID.

SNVT Reference

SNVT_setting #117

4.4.6. **Setting My/Stop**

```
network input config UCPTsettingMy nciSettingMy
```

Values which are sent via nvoSetting if My command has been received.

Valid Range

The valid range of SNVT_setting.

Default Value

SET_NUL, INVALID, INVALID.SNVT Reference

SNVT_setting #117

4.4.7. **Step Angle**

```
network input config UCPTstepAngle nciStepAngle
```

Angle which is used in setting to tilt blinds up or down if Long Operation time was not exceeded.

Valid Range

The valid range is 0 to 180,0 degrees.

Default Value

The default value is 0,0 degrees.

Configuration Requirements/Restrictions

The physical capabilities of the blinds and the internal solution of tilt range determine the smallest possible tilt step angle.

SNVT Reference

SNVT_angle_deg #104

4.4.8. **Scroll Step**

```
network input config UCPTscrollStep nciScrollStep
```

Percentage of value change for tilting and dimming if a scroll message has been received. For tilting the percentage concerns the range of 180°.

Valid Range

The valid range is 0,0 to 100,0 percent.

Default Value

The default value is 10,0 percent.

Configuration Requirements/Restrictions

Depending on the function entered in nciSettingUp and nciSettingDown scrolling performs an absolute tilting (function = SET_STATE) or a relative one (any other function).

The physical capabilities of the blinds and the internal solution of tilt range determine the smallest possible tilt step angle.

SNVT Reference

SNVT_level_cont #21

Configuration Considarations

Only valid with Telis Modulis (wheel) TILTING and Telis (long press) DIMMING
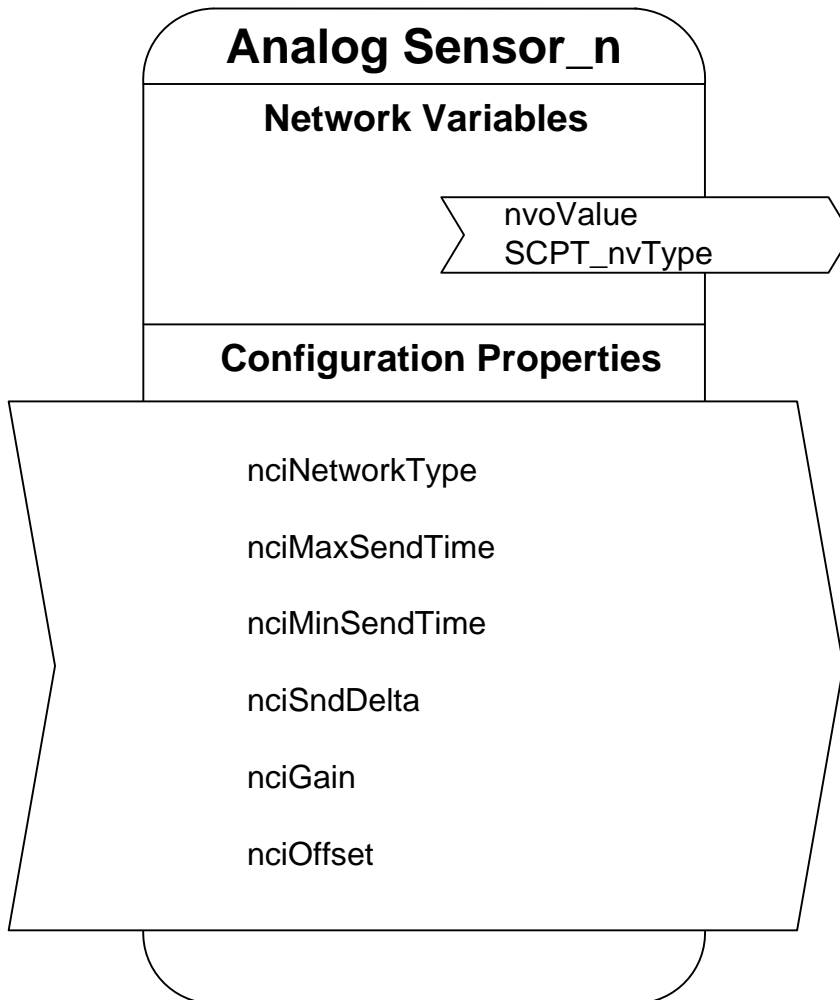
See also chapter 6. Appendix

5. Analog Sensor #0001

5.1. Overview

The Analog Sensor inherits all mandatory and some optional members from the LonMark[®] Analog Sensor Object # 0001. Various SOMFY specific members have been added.

The Functional Block "Analog Sensor" exists 3 times in the animeo LON RTS Module. Each could be set individually and separately from the others.

5.2. Functional-Block Details

```
┌──────────────────────────────────┐
│         Analog Sensor_n          │
├──────────────────────────────────┤
│        Network Variables         │
│                                  │
│              nvoValue            │
│             SCPT_nvType          │
├──────────────────────────────────┤
│     Configuration Properties     │
│                                  │
│          nciNetworkType          │
│                                  │
│          nciMaxSendTime          │
│                                  │
│          nciMinSendTime          │
│                                  │
│           nciSndDelta            │
│                                  │
│             nciGain              │
│                                  │
│            nciOffset             │
└──────────────────────────────────┘
```

*   The Functional Block "Analog Sensor" exists 3 times at the animeo LON RTS Module.
    Each could be set individually and separately from the others.

Table 1 SNVT Details

NV (M/O)*	Variable Name	SNVT Name	Description
1(M)	nvoValue_x	changeable	Value of used Analog sensor

* M = mandatory, O = optional

Table 2 SCPT Details

(M/O)**	SCPT Name	SCPT Index	Associated NVs**	Description
(O)	SCPTnvType nciNetworkType SNVT_nv_type (166)	254	nv1	Defines used type of Analog sensor (e. g. Global Radiation, CO2, Temperature, Brightness, Occupancy, ...)
(O)	SCPTmaxSendTime nciMaxSendTime SNVT_time_sec (107)	49	nv1	Maximum period of time that expires before the object will automatically update the NV
(O)	SCPTminSendTime nciMinSendTime SNVT_time_sec (107)	52	nv1	Minimum period of time that expires before the object will automatically update the NV
(O)	SCPTsndDelta nciSndDelta SNVT inherited	27	nv1	Sets the change in value that must result before the Object updates the NV
(O)	SCPTgain nciGain SNVT_multidiv (91)	31	nv1	Defines the gain of nvoAnaolgSensorOut
(O)	SCPToffset nciOffset SNVT inherited	26	nv1	Defines the offset of nvoAnaolgSensorOut

* M = mandatory, O = optional
It should be mandatory for CPs that mandatory for an NV that is also mandatory. This is also valuable for CPs that applies to the Entire Functional Block.

** List of NVs to which this configuration property applies.
An "(M)" means that the CP is mandatory if the NV (to which it applies) is implemented.
An "(O)" means that the CP is optional if the NV (to which it applies) is implemented.

5.3. Network Variables

5.3.1. Analog Sensor Output x

```
network output SNVT(SCPTnvType) nvoValue_x;
```

This output network variable reports the value of the sensor which is connected via 12-bit A/D-converter to the corresponding analog input (all one- and two-byte integer values plus SNVT_switch and SNVT_occupancy are supported).

The hardware is designed to accept 0 ... 10 V DC.

Valid Range

Depends on selected NV-type, raw 0 … FFFF

Default Value

0

Configuration Considerations

Behaviour depends on the values of the following properties:

SCPTnvType

SCPTmaxSendTime

SCPTminSendTime

SCPTsndDelta

SCPTgain

SCPToffset

5.4. Configuration Properties

5.4.1. Type Of Analog Sensor

```
network input config SCPTnvType nciNvType;
```

Valid Range

nvoValue of following type categories are supported:

- NVT_CAT_UNSIGNED_SHORT e. g. SNVT_lev_cont

- NVT_CAT_SIGNED_LONG e. g. SNVT_temp

- NVT_CAT_UNSIGNED_LONG e. g. SNVT_Wm2_p

- NVT_CAT_STRUCT: **only** SNVT_switch

- NVT_CAT_ENUM: **only** SNVT_occupancy

Default Value

SNVT_lux,

PID 0:0:0:0:0:0:0:0, Scope 0, Index 79, NVT_CAT_UNSIGNED_LONG, 2 bytes, A=1, B=0, C=0

SCPT Reference

SCPTnvType (254)

Configuration Considerations

The type can only be changed if the nvoValue_x is not bound to any other NV. If this is not observed or if an unsupported type is set the device goes offline and restores the previous setting.

5.4.1.1. Example: CO2 As Analog Sensor

```
network input config SCPTnvType nciNetworkType =
SNVT_lev_percent;
```

This input configuration property sets the carbon dioxide (CO^2) value in parts-per-million (ppm) at the following network variables:

- nv1 – nvoValue_x

Valid Range of Analog Sensor Output

-163.840 … 163.835 ppm (Resolution 0.001ppm)

Default Value of Analog Sensor Output

0 ppm (parts-per-million)

5.4.1.2. Example: Temperature As Analog Sensor

```
network input config SCPTnvType nciNetworkType =
SNVT_temp_p;
```

This input configuration property sets the temperature value in degree Celsius (°C) at the following network variables:

- nv1 – nvoValue_x

Valid Range of Analog Sensor Output

-273.17 … 327.67°C (Resolution 0.01°C)

Default Value of Analog Sensor Output

0 °C (Degree Celsius)

5.4.1.3. Example: Brightness As Analog Sensor

```
network input config SCPTnvType nciNetworkType = SNVT_lux;
```

This input configuration property sets the brightness value in lux at the following network variables:

- nv1 – nvoValue_x

Valid Range of Analog Sensor Output

0 … 65535 lx (Resolution 1.0 lx)

Default Value of Analog Sensor Output

0 lx

5.4.1.4. Example: Occupancy As Dry Contact

```
network input config SCPTnvType nciNetworkType =
SNVT_occupancy;
```

This input configuration property sets the voltage level to occupancy at the following network variables:

- nv1 – nvoValue_x

Valid Range of Analog Sensor Output

0 = OC_OCCUPIED

1 = OC_UNOCCUPIED

Default Value of Analog Sensor Output

-1 = OC_NUL

5.4.1.5. **Example: Watts per square meter As Analog Sensor**

```
network input config SCPTnvType nciNetworkType =
SNVT_Wm2_p
```

This input configuration property sets the global radiation value in W/m² at the following network variables:

- nv1 – nvoValue_x

Valid Range of Analog Sensor Output

0 … 3276,70 W/m² (Resolution 0.05W/m²)

Default Value of Analog Sensor Output

0 W/m²

5.4.2. **Max Send Time (Heartbeat)**

```
network input config SNVT_time_sec nciMaxSendTime;
```

This input configuration property sets the maximum period of time that can expire before the Object will automatically (cyclically) update one of the following network variables:

- nv1 – nvoValue_x

Valid Range

0 … 6553.5 seconds (only complete seconds will be processed)

Default Value

0.0 seconds (no automatic update).

SCPT Reference

SCPTmaxSendTime (49)

5.4.3. **Min Send Time**

```
network input config SNVT_time_sec nciMinSendTime;
```

This input configuration property sets the minimum period of time that must expire before the Object will automatically (cyclically) update one of the following network variables:

- nv1 – nvoValue_x

The internal measuring hardware is controlled as well by this configuration property, Zero means measuring disabled.

Valid Range

0 …. 6553.5 seconds (only complete seconds will be processed)

Default Value

0.0 seconds.

SCPT Reference

SCPTminSendTime (52)

Configuration Considerations

Reset Flag set: The device should be reset after modification of this configuration property

MinSendTime has to be set to 1.0 seconds to be active.

0 = inactive

5.4.4. **Send On Analog Sensor Delta**

SCPT Reference

SCPTsndDelta (27)

5.4.4.1. Example: Send On Power Delta

```
network input config SNVT_power nciSndDelta;
```

This input configuration property sets the change of value in Watt that must result before the Object will automatically update the following network variables:

- nv1 – nvoValue_x

Valid Range

0,0 … 6553,5W (Resolution 0.1W)

Default Value

10 W (Watt)

5.4.4.2. Example: Send On Percent (CO2) Delta

```
network input config SNVT_lev_percent nciSndDelta;
```

This input configuration property sets the change of value in percent that must result before the Object will automatically update the following network variables:

- nv1 – nvoValue_x

Valid Range

-163.840 … 163.835 ppm (Resolution 0.005ppm)

Default Value

500 ppm (parts-per-million)

5.4.4.3. Example: Send On Temperatur Delta

```
network input config SNVT_temp_p nciSndDelta;
```

This input configuration property sets the change of value in °C that must result before the Object will automatically update the following network variables:

- nv1 – nvoValue_x

Valid Range

-273.17 … 327.67°C (Resolution 0.01°C)

Default Value

1 °C (Degree Celsius)

5.4.4.4. Example: Send On Brightness Delta

```
network input config SNVT_lux nciSndDelta;
```

This input configuration property sets the change of value in Lux that must result before the Object will automatically update the following network variables:

- nv1 – nvoValue_x

Valid Range

0.0 … 65535 lx (Resolution 1.0 lx)

Default Value

100 lx (Lux)

5.4.4.5. Example: Send On Switch Delta

```
network input config SNVT_switch nciSndDelta;
```

This input configuration property sets the change of value in percent that must result before the Object will automatically update the following network variables:

- nv1 – nvoValue_x

Valid Range

The SNVT_switch. State is ignored. The range of the 8-bit intensity value is 0 - 200 (0 - 100% in 0,5% steps). Minimum level is value 0.

Default Value

The default value is value = 0,0, state = 0.

5.4.4.6. Example: Send On Occupancy Delta

Not applicable. Update occurs on status change.

5.4.4.7. Example: Watts per square meter As Analog Sensor

```
network input config SCPTnvType nciNetworkType =
SNVT_Wm2_p
```

This input configuration property sets the change of value in W/m² that must result
before the Object will automatically update the following network variables:

- nv1 – nvoValue_x

Valid Range of Analog Sensor Output

0 … 3276,70 W/m² (Resolution 0.05W/m²)

Default Value of Analog Sensor Output

0 W/m²

5.4.5. **Analog Sensor Gain**

```
network input config SNVT_multidiv nciGain;
```

This input configuration property defines a linear sensor in combination with NV
nciGain_[x] in range of the following network variables:

- nv1 – nvoValue_x

Multiplier defines the measurement range between minimum, maximum and used
resolution.

Divisor defines the resolution of the used A/D converter, around 2122 for 10 V

Structure Definition

```
typedef struct {
          unsigned long multiplier;
          unsigned long divisor;
} SNVT_muldiv;
```

Valid Range

Multiplier = 0 … 65535 (Resolution 1)

Divisor = 1 … 65535 (Resolution 1)

Default Value

Multiplier = 1

Divisor = 1

SCPT Reference

SCPTgain (31)

E.g. you would like to use a brightness sensor (sun sensor) with a measurement range from 0 lux to 100000 lux and a resolution with 1 lux
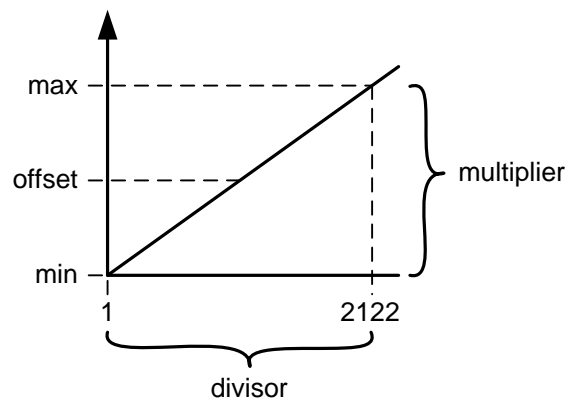
=> *multiplier* = 100000

But the maximum value of multiplier can only be 65535. Therefore a variable is needed which shows the same ratio between *multiplier* and *divisor*; e.g.

$$\frac{\text{multiplier}}{\text{divisor}} = \frac{100000}{2122} \approx \frac{65535}{1391}$$

multiplier => 65535

divisor => 1391

max ┄┄

offset ┄┄   } multiplier

min ┄┄

1   2122

}
divisor

5.4.5.1. Example: Electric Voltage As Analog Sensor

```
network input config SCPTnvType nciNetworkType =
SNVT_volt;
```

This input configuration property sets the electric voltage value in volt at the following network variables:

- nv1 – nvoValue_x

Theoretical Range of SNVT_volt

-10,0 … 10,0 Volt (Resolution 0,1 V)

Multiplier

20,0 / 0,1 = 200

5.4.5.2. Example: Brightness As Analog Sensor

```
network input config SCPTnvType nciNetworkType = SNVT_lux;
```

This input configuration property sets the brightness value into lux at the following network variables:

- nv1 – nvoValue_x

Theoretical Range of SNVT_lux

0 … 100000 lx (Resolution 1,0 lx)

Attention: nvoValue can only output up to 65535 !

Multiplier

65535 / 1,0 = 65535

5.4.5.3. Example: Occupancy Or Switch As Dry Contact

```
network input config SCPTnvType nciNetworkType =
SNVT_occupancy or SNVT_switch;
```

Multiplier = Devisor

5.4.6. Analog Sensor Offset

SCPT Reference

SCPToffset (26)

5.4.6.1. Example: Analog Sensor Offset Power

```
network input config SNVT_power nciOffset;
```

This input configuration property defines a linear global radiation sensor in combination with NV nciGain_[x] in range of the following network variables:

- nv1 – nvoValue_x

Valid Range

0.0 … 6553.5 W (Resolution 0.1W)

Default Value

0 W (Watt)

5.4.6.2. Example: Analog Sensor Offset CO2

```
network input config SNVT_lev_percent nciOffset;
```

This input configuration property defines a linear CO2 sensor in combination with NV nciGain_[x] in range of the following network variables:

- nv1 – nvoValue_x

Valid Range

-163.840 … 163.835 ppm (Resolution 0.005 ppm)

Default Value

0 ppm (parts-per-million)

5.4.6.3. Example: Analog Sensor Offset Temperature

```
network input config SNVT_temp_p nciOffset;
```

This input configuration property defines a linear temperature sensor in combination with NV nciGain_[x] in range of the following network variables:

- nv1 – nvoValue_x

Valid Range

-273.17 … 327.67°C (Resolution 0.01°C)

Default Value

0°C (Degree Celsius)

5.4.6.4. Example: Analog Sensor Offset Brightness

```
network input config SNVT_lux nciOffset;
```

This input configuration property defines a linear brightness sensor in combination with NV nciGain_[x] in range of the following network variables:

- nv1 – nvoValue_x

Valid Range

0 … 65535 lx (Resolution 1.0 lx)

Default Value

0 lx

5.4.6.5. Example: Analog Sensor Offset Switch

```
network input config SNVT_switch nciOffset;
```

- This input configuration property defines a switch sensor in combination with NV nciGain_[x] in range of the following network variables:
- nv1 – nvoValue_x

Valid Range

The SNVT_switch. State is ignored. The range of the 8-bit intensity value is 0 - 200 (0 - 100% in 0,5% steps). Minimum level is value 0.

Default Value

The default value is value = 0,0, state = 0.

5.4.6.6. Example: Analog Sensor Offset Occupancy

```
network input config SNVT_occupancy nciOffset;
```

This input configuration property defines a occupancy sensor in combination with NV nciGain_[x] in range of the following network variables:

- nv1 – nvoValue_x

Valid Range

0 = OC_OCCUPIED: normally closed dry contact

1 = OC_UNOCCUPIED: normally open dry contact

Default Value

0 =   OC_OCCUPIED

5.4.6.7. Example: Watts per square meter As Analog Sensor

```
network input config SCPTnvType nciNetworkType =
SNVT_Wm2_p
```

This input configuration property sets the global radiation value into W/m² at the following network variables:

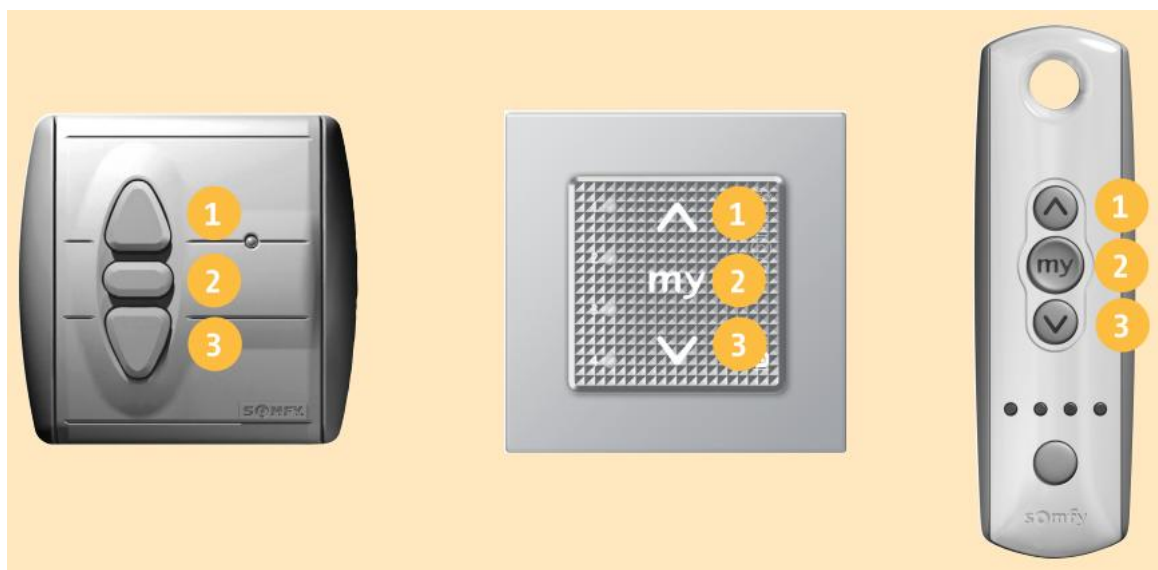- nv1 – nvoValue_x

Valid Range of Analog Sensor Output

0 … 3276,70 W/m² (Resolution 0.05W/m²)

Default Value of Analog Sensor Output

0 W/m²

6. Appendix:

6.1. Push button configuration of the radio transmitter

				Blinds		Dimming	
1	UP button	Long press	Move	SettingUp	Brighter	Scroll_Step
		Short press	Tilt	Step_Angle	On	Switch_On
2	"my" button	Long press	Move	Setting my	Toggle On/ Off	
		Short press	Stop	Setting Stop		
3	DOWN button	Short press	Tilt	Step_Angle	Off	Switch_Off
		Long press	Move	Setting_Down	Darker	Scroll_Step

6.2. Push button configuration of the radio transmitter

			Blinds		Dimming	
1	UP button	Long and short press	Move	Setting_Up	On	Switch On
2	"my" button	Long press	Move	Setting_my	Toggle On/ Off	
		Short press	Stop	Setting_Stop		
3	DOWN button	Long and short press	Move	Setting_Down	Off	Switch Off
4	Scroll wheel	Turn Up and Down	Tilt	Scroll Step	Brighter/ Darker	Scroll Step

6.3. SOMFY RTS Transmitter compatible with

6.3.1. **LON RTS Module 433 MHz; #1860239**

The following Somfy RTS remotes with a frequency of 433 MHz are compatible with the LON RTS Module 433 MHz WM, Ref. 1860239:

Telis 1 RTS 433 MHz
Telis 4 RTS 433 MHz
Telis 16 RTS 433 MHz
Telis 6 Chronis RTS 433 MHz
Telis 1 Modulis RTS 433 MHz
Telis 4 Modulis RTS 433 MHz
Chronis RTS L 433 MHz
Chronis RTS Smart 433 MHz
Smoove 1 RTS 433 MHz
Smoove Origin RTS 433 MHz
Situo RTS 433 MHz

6.3.2. **LON RTS Module 447 MHz; #1860240**

The following Somfy RTS remotes with a frequency of 447 MHz are compatible with the LON RTS Module 447 MHz WM, Ref. 1860240:

Telis 1 RTS 447 MHz
Telis 4 RTS 447 MHz
Telis 16 RTS 447 MHz
Telis 6 Chronis RTS 447 MHz
Telis 4 Modulis RTS 447 MHz
Situo RTS Pure 447 MHz