Functional Profile

SOMFY animeo®
LON Sensor Interface WM

Ref. 1860161

Version 1.1
Revision date: 16.12.09

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		23.10.2009
1.1	Update		16.12.2009

2. Node Object #0000– Outside Sensor Box (OSB)

2.1. Functional-Block Details

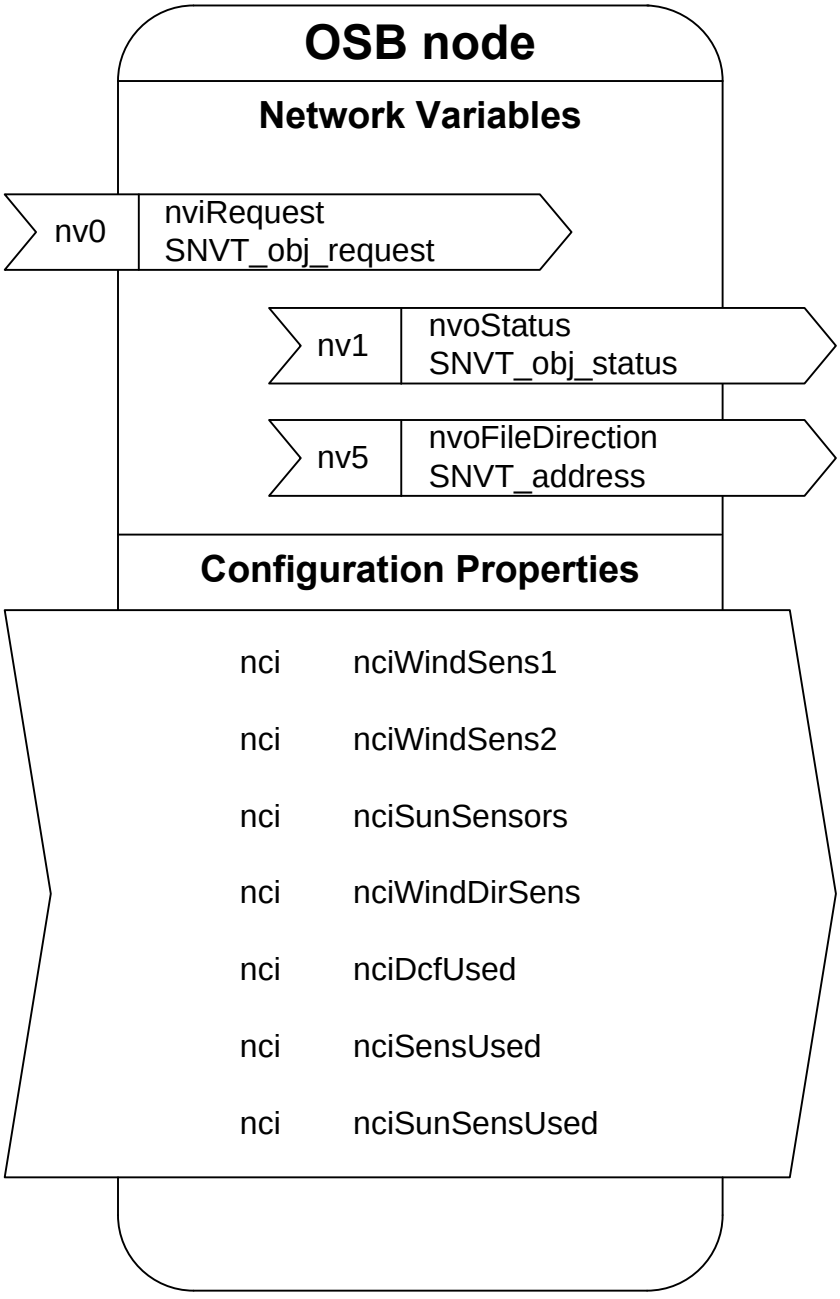


Table 1 SNVT Details

NV (M/O)*	Variable Name	SNVT Name	SNVT 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
5(O)	nvoFileDirectory	SNVT_address	114	Address for the file directory containing descriptors for configuration files

* M = mandatory, O = optional

Table 2 SCPT Details

(M/O) **	SCPT/UCPT Name	SCPT/ UCPT Index	Associated NVs**	Description
(O)	UCPTwindSens1 nciWindSens1		Entire Device	Defines the type of connected wind sensor 1
(O)	UCPTwindSens2 nciWindSens2		Entire Device	Defines the type of connected wind sensor 2
(O)	UCPTsunSensors nciSunSensors		Entire Device	Defines the type of connected sun sensor(s)
(O)	UCPTwindDirSens nciWindDirSens		Entire Device	Defines the type of connected wind direction sensor
(O)	UCPTdcfUsed nciDcfUsed		Entire Device	Defines whether DCF-time/date signal shall be used or not
(O)	UCPTsensUsed nciSensUsed		Entire Device	Defines which sensors are connected and used at the Outside Sensor Box and at the 3 Analog Inputs
(O)	UCPTsunSensUsed nciSunSensUsed		Entire Device	Defines which sun sensors are connected and used at the Outside Sensor Box

* 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.

2.2. Network Variables

2.2.1. 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.

2.2.2. 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

The optional **nciMaxStsSendT** configuration property specifies a heartbeat for sending this network variable. If the CP is not implemented, or is implemented and is set to zero or the invalid value, a heartbeat is not provided.

When Transmitted

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

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

The heartbeat interval specified by the optional **nciMaxStstSendT** CP expires.

When the heartbeat timer expires, the status of each functional block (including the Node Object functional block) is returned sequentially in round-robin fashion—one object status per expiration of the timer.

Default Service Type

The default service type is acknowledged.

2.2.3. 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.

2.3. Configuration Properties

2.3.1. Wind Sensor Type

```
network input config UCPTwindSens nciWindSens[x];
```

This input configuration property sets the type of used wind sensor which is connected to the wind sensor input 1 and or wind sensor input2

Valid Range

Value	Identifier	Notes
-1	MEM_NULL	Not defined
0	NO_SENSOR	No Sensor
17	WIND_BIG_HEATED	Heated Wind Sensor (Ref. 9140180)
18	WIND_SMALL	Wind Sensor (Ref. 9127932)
32	WIND_BIG	Wind Sensor (not heated) (Ref. 9001608)
48	WIND_THIES	Wind Sensor (Thies)

Default Value

17 = Heated Wind Sensor (Ref. 9140180)

SCPT Reference

UCPTwindSens[x]

2.3.2. Sun Sensor Type

```
network input config UCPTsunSensors nciSunSensors;
```

This input configuration property sets the type of used sun sensor which is connected to the sun sensor input 1, input2, input3, input4, input 5, input6, input7 and input8.

Valid Range

Value	Identifier	Notes
-1	MEM_NULL	Not defined
16	SOMFY	SOMFY sun sensor type (Ref. 9050100)

Default Value

16 = SOMFY sun sensor type (Ref. 9050100)

SCPT Reference

UCPTsunSensors

2.3.3. Wind Direction Sensor Type

```
network input config UCPTwindDirSens nciWindDirSens;
```

This input configuration property sets the type of used wind sensor which is connected to the wind direction sensor .

Valid Range

Value	Identifier	Notes
-1	MEM_NULL	Not defined
0	NO_WINDDIR_ SENSOR	No Sensor
32	WINDDIR_SOMFY	Wind Direction Sensor (SOMFY new Ref.9013807 or old Ref.9001609)
48	WINDDIR_THIES	Wind Direction Sensor (Thies)

Default Value

32 = Wind Direction Sensor (SOMFY new Ref.9013807 or old Ref.9001609)

SCPT Reference

UCPTwindDirSens

2.3.4. DCF Modul Configuration

```
network input config UCPTdcfUsed nciDcfUsed;
```

This input configuration property sets the usage of a DCF Receiver which is plugged in at the Outside Sensor Box.

Valid Range

Value	Identifier	Notes
-1	MEM_NULL	Not defined
0	NO_SENSOR	No Sensor
16	DCF_SOMFY	DCF Receiver (Ref.9001612)

Default Value

16 = DCF Receiver (Ref.9001612)

SCPT Reference

UCPTdcfUsed

2.3.5. Sensors connected

```
network input config UCPTsensUsed1 nciSensUsed;
```

This input configuration property defines which sensors are connected and used at the Outside Sensor Box and at the Analog Inputs 1 to 3. First bit sets the Wind Sensor_1, second Wind Sensor_2, third Wind Direction Sensor, fourth Temperature, fifth Rain, sixth AnalogSensorOut_1, seventh AnalogSensorOut_2 and eighth AnalogSensorOut_3.

Valid Range

Value	Identifier	Notes
0	NO_SENSOR	Sensor not connected/ not used
1	SENSOR	Sensor connected/used

Default Value

0,0,0,0,0,0,0,1 (extSens3=NO, extSens2=NO; extSens1=NO; rain=NO; temp=NO; windDir=NO, wind2=NO, wind1=YES,)

SCPT Reference

UCPTsensUsed1

2.3.6. Sun Sensors connected

```
network input config UCPTsunSensUsed nciSunSensUsed;
```

This input configuration property defines which Sun Sensors are connected and used at the Outside Sensor Box.

Valid Range

Value	Identifier	Notes
0	NO_SENSOR	Sensor not connected/ not used
1	SENSOR	Sensor connected/used

Default Value

0,0,0,0,0,0,0,1 (Sun_8=NO, Sun_7=NO, Sun_6=NO, Sun_5=NO, Sun_4=NO, Sun_3=NO, Sun_2=NO, sun_1=YES)

SCPT Reference

UCPTsunSensorUsed

3. Rain Sensor #1051

3.1. Functional-Block Details

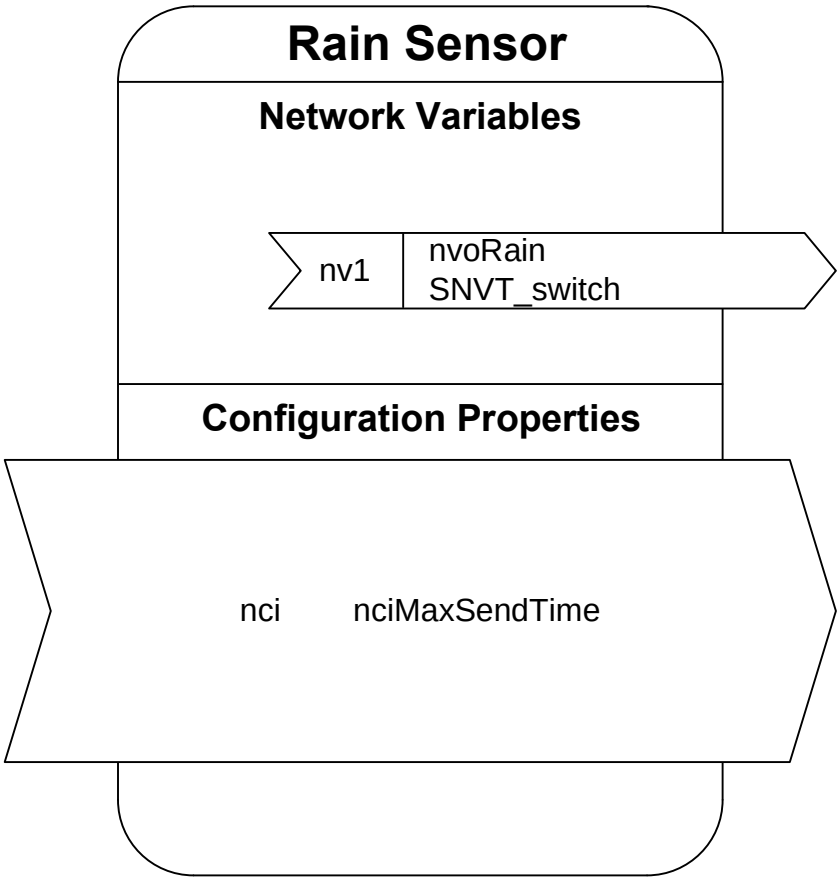


Table 1 SNVT Details

NV (M/O)*	Variable Name	SNVT Name	SNVT Index	Description
1(M)	nvoRain	SNVT_switch	95	Rain sensor output

* M = mandatory, O = optional

Table 2 SCPT Details

(M/O) **	SCPT Name	SCPT Index	Associated NVs**	Description
(O)	SCPTmaxSendTime nciMaxSendTime SNVT_time_sec(107)	49	nv1	Maximum period of time that expires before the object will automatically update NV

* 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.

3.2. Network Variables

3.2.1. Rain Output

```
network input SNVT_switch nvoRain;
```

This output network variable is used to send rain (precipitation) sensor influence on the controller.

Valid Range

For details refer to the *LONMARK SNVT Master List*, versions 13.00 and later.

nviRain.state	nviRain.value	Meaning
0	0	no rain
1	0	no rain
1	1..200 (0.5% .. 100.0%)	rain level
0xFF	not considered	INVALID

Default Value

nviRain.value = 0

nviRain.state = 0xFF

Configuration Considerations

Behaviour depends on the values of the following properties:

SCPTmaxSendTime

3.3. Configuration Properties

3.3.1. Max Send Time

```
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 variable:

nv1 – nvoRain

Valid Range

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

Default Value

60.0 seconds.

SCPT Reference

Remark: Since the LonMark Standard #1051 uses the wrong reference (#52) this CP appears as SCPTminSendTime.

SCPTmaxSendTime (49)

4. Real Time Keeper #3300

4.1. Functional-Block Details

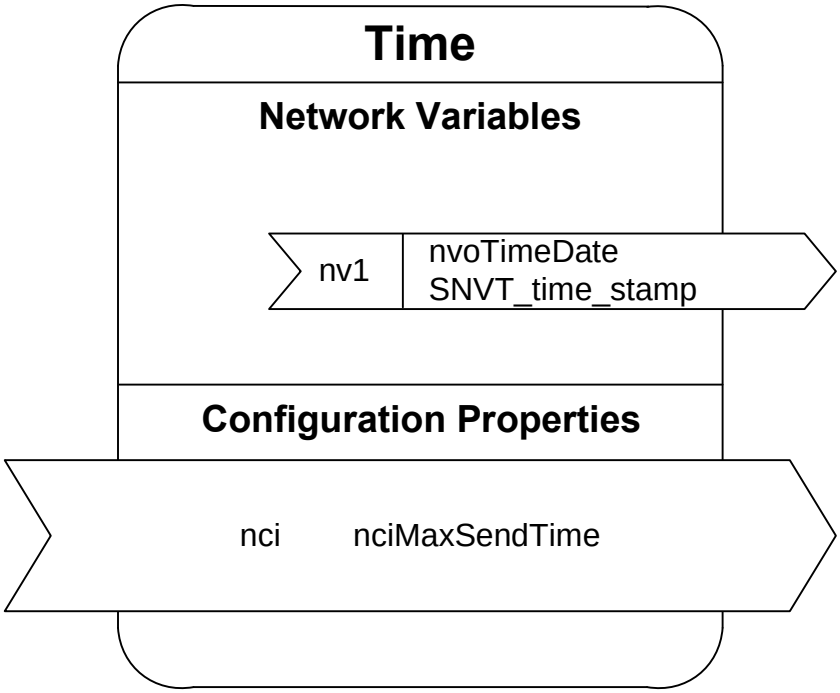


Table 1 SNVT Details

NV (M/O)*	Variable Name	SNVT Name	SNVT Index	Description
1(M)	nvoTimeDate	SNVT_time_stamp	84	Actual time and date output

* M = mandatory, O = optional

Table 2 SCPT Details

(M/O) **	SCPT Name	SCPT Index	Associated NVs**	Description
(M)	SCPTmaxSendTime nciMaxSendTime SNVT_time_sec(107)	49	nv1	Maximum period of time that expires before the object will automatically update NV

* 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.

4.2. Network Variables

4.2.1. Time Date Output

```
network output SNVT_time_stamp  nvoTimeDate;
```

This output network variable reports time and date.

Valid Range

Year: 0 .. 3000

Month: 0 .. 12

Day: 0 .. 31

Hour: 0 .. 23

Minute: 0 .. 59

Second: 0 .. 59 (Resolution 1)

Default Value

Year: 2009

Month: 01

Day: 01

Hour: 12

Minute: 00

Second: 00

Configuration Considerations

Behaviour depends on the values of the following properties:

SCPTmaxSendTime

4.3. Configuration Properties

4.3.1. Max Send Time

```
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 variable:

nv1 – nvoTimeDate

Valid Range

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

Default Value

3600.0 seconds.

SCPT Reference

SCPTmaxSendTime (49)

5. Wind direction #20006

5.1. Functional-Block Details

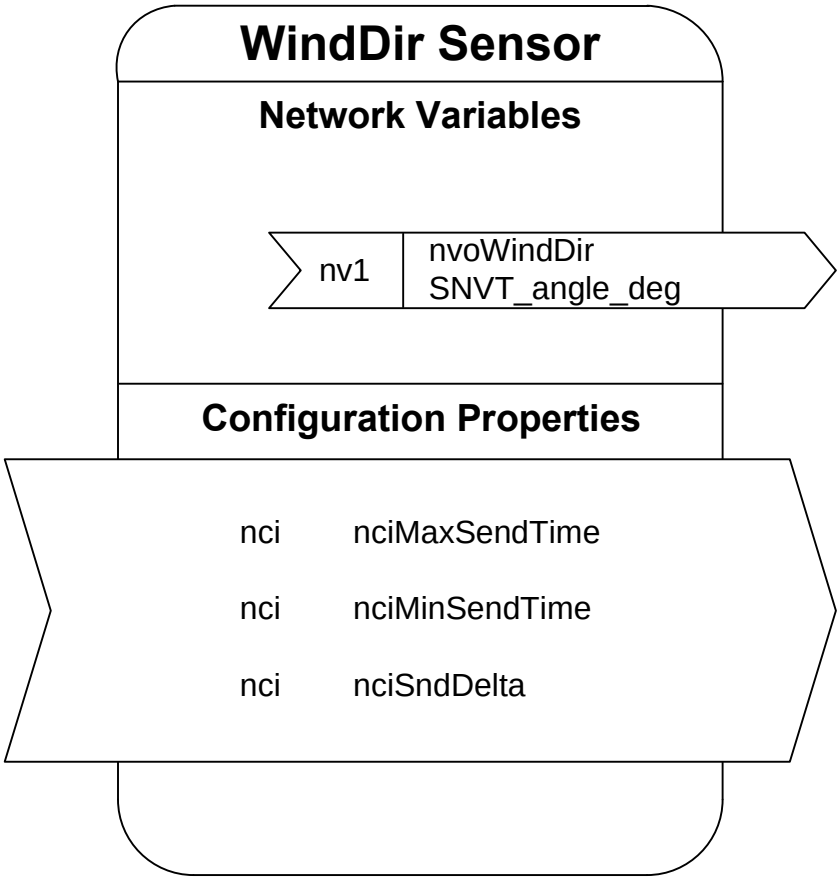


Table 1 SNVT Details

NV (M/O)*	Variable Name	SNVT Name	SNVT Index	Description
1(M)	nvoWindDir	SNVT_angle_deg	104	Wind direction sensor output

* M = mandatory, O = optional

Table 2 SCPT Details

(M/O) **	SCPT Name	SCPT Index	Associated NVs**	Description
(M)	SCPTmaxSendTime nciMaxSendTime SNVT_time_sec(107)	49	nv1	Maximum period of time that expires before the object will automatically update NV
(M)	SCPTminSendTime nciMinSendTime SNVT_time_sec(107)	52	nv1	Minimum period of time that expires before the object will automatically update NV
(M)	SCPTsndDelta nciSndDelta SNVT_angle_deg(104)	27	nv1	Sets the minimum change required to force transmission of the output value

* 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.

5.2. Network Variables

5.2.1. Wind Direction Output

```
network output SNVT_Speed  nvoWindDir;
```

This output network variable reports wind direction with a range from -359.98 …360.00° (Degree)

Valid Range

-359.98…360.00° (Resolution 0.01°)

Default Value

180°.

Configuration Considerations

Behaviour depends on the values of the following properties:

SCPTmaxSendTime

SCPTminSendTime

SCPTsndDelta

5.3. Configuration Properties

5.3.1. Max Send Time

```
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 variable:

- nv1 – nvoWindDir

Valid Range

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

Default Value

60.0 seconds.

SCPT Reference

SCPTmaxSendTime (49)

5.3.2. 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 variable:

- nv1 – nvoWindDir

Valid Range

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

Default Value

10.0 seconds.

SCPT Reference

SCPTminSendTime (52)

5.3.3. Send On Delta Value

```
network input config SNVT_angle_deg nciSndDelta;
```

This input configuration property sets the change in value that must result before the Object will automatically (cyclically) update the following network variable:

- nv1 – nvoOutdoorTemp

Valid Range

-273.17 ... 327.66°C (Resolution 0.01°C)

Default Value

90°C (Degree Celsius)

SCPT Reference

SCPTsndDelta (26)

6. Outside Temperature Sensor #1040

6.1. Functional-Block Details

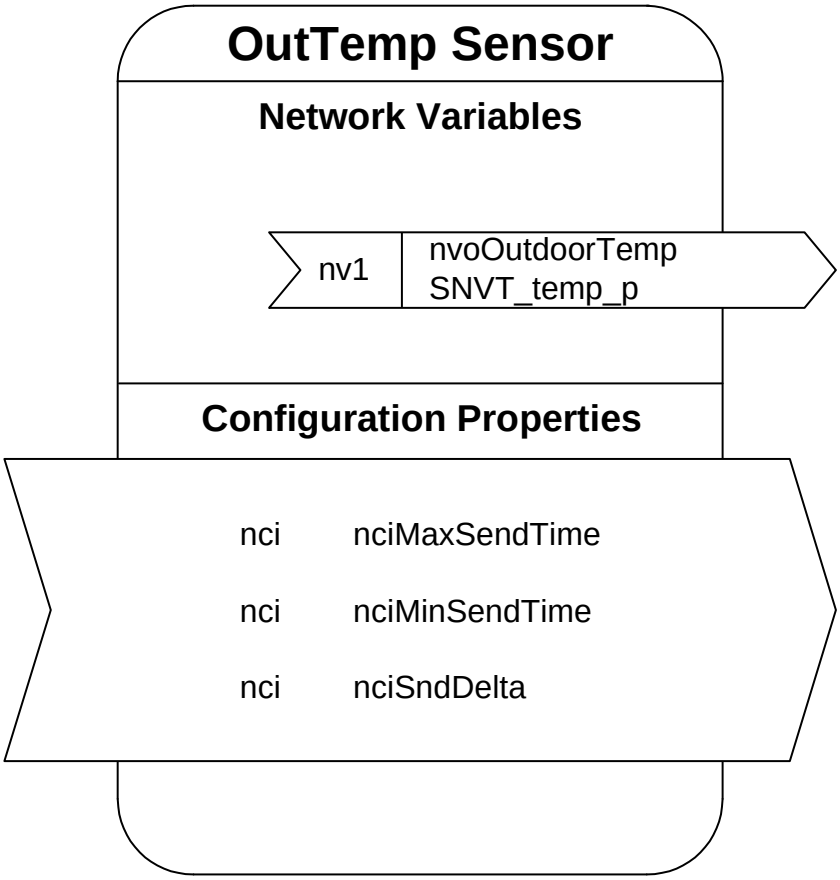


Table 1 SNVT Details

NV (M/O)*	Variable Name	SNVT Name	SNVT Index	Description
1(M)	nvoOutdoorTemp	SNVT_temp_p	105	Outdoor air temperature output

* M = mandatory, O = optional

Table 2 SCPT Details

(M/O) **	SCPT Name	SCPT Index	Associated NVs**	Description
(M)	SCPTmaxSendTime nciMaxSendTime SNVT_time_sec(107)	49	nv1	Maximum period of time that expires before the object will automatically update NV
(M)	SCPTminSendTime nciMinSendTime SNVT_time_sec(107)	52	nv1	Minimum period of time that expires before the object will automatically update NV
(M)	SCPTminDeltaTemp nciMinDelta SNVT_temp_p (105)	64	nv1	Sets the minimum change required to force transmission of the output value

* 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.

6.2. Network Variables

6.2.1. Outdoor Temperature Output

```
network output SNVT_temp_p  nvoOutdoorTemp;
```

This output network variable reports outdoor Temperature with a range from -273.17…327.66°C (Degree Celsius).

Valid Range

-273.17…327.66°C (Resolution 0.01°C)

Default Value

0 °C.

Configuration Considerations

Behaviour depends on the values of the following properties:

SCPTmaxSendTime

SCPTminSendTime

SCPTminDeltaTemp

6.3. Configuration Properties

6.3.1. Max Send Time

```
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 the following network variable:

- nv1 – nvoOutdoorTemp

Valid Range

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

Default Value

60.0 seconds.

SCPT Reference

SCPTmaxSendTime (49)

6.3.2. 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 the following network variable:

- nv1 – nvoOutdoorTemp

Valid Range

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

Default Value

10.0 seconds.

SCPT Reference

SCPTminSendTime (52)

6.3.3. Send On Delta Value

```
network input config SNVT_temp_p nciMinDelta;
```

This input configuration property sets the change in value that must result before the Object will automatically (cyclically) update the following network variable:

- nv1 – nvoOutdoorTemp

Valid Range

-273.17 .. 327.66°C (Resolution 0.01°C)

Default Value

5°C (Degree Celsius)

SCPT Reference

SCPTminDeltaTemp (64)

7. Sun Sensors #1010

7.1. Functional-Block Details

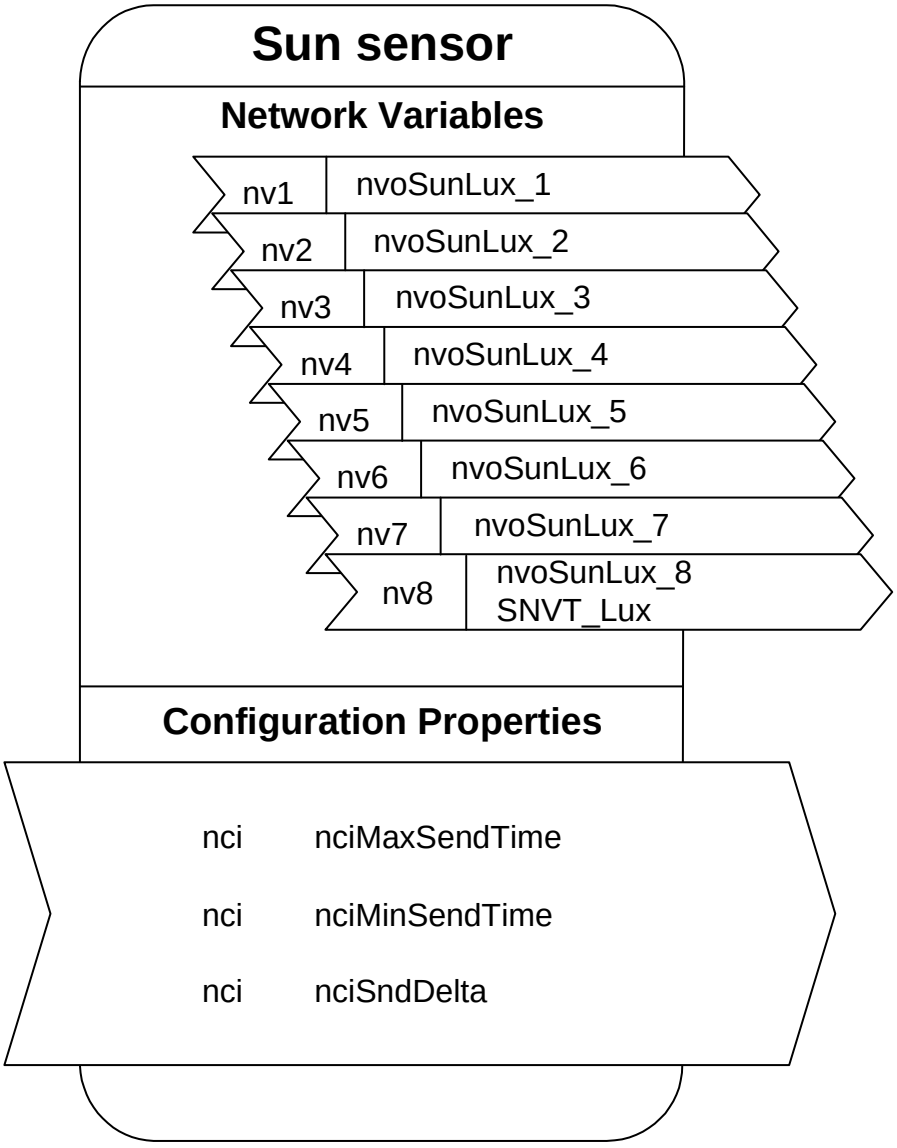


Table 1 SNVT Details

NV (M/O)*	Variable Name	SNVT Name	SNVT Index	Description
1(M)	nvoSunLux_1	SNVT_lux	79	Sun (outdoor) brightness output
2(M)	nvoSunLux_2	SNVT_lux	79	Sun (outdoor) brightness output
3(M)	nvoSunLux_3	SNVT_lux	79	Sun (outdoor) brightness output
4(M)	nvoSunLux_4	SNVT_lux	79	Sun (outdoor) brightness output
5(M)	nvoSunLux_5	SNVT_lux	79	Sun (outdoor) brightness output
6(M)	nvoSunLux_6	SNVT_lux	79	Sun (outdoor) brightness output
7(M)	nvoSunLux_7	SNVT_lux	79	Sun (outdoor) brightness output
8(M)	nvoSunLux_8	SNVT_lux	79	Sun (outdoor) brightness output

* M = mandatory, O = optional

Table 2 SCPT Details

(M/O) **	SCPT Name	SCPT/UCPT Index	Associated NVs**	Description
(M)	SCPTmaxSendTime nciMaxSendTime SNVT_time_sec(107)	49	nv1, nv2, nv3, nv4, nv5, nv6, nv7, nv8	Maximum period of time that expires before the object will automatically update NV
(M)	SCPTminSendTime nciMinSendTime SNVT_time_sec(107)	52	nv1, nv2, nv3, nv4, nv5, nv6, nv7, nv8	Minimum period of time that expires before the object will automatically update NV
(M)	SCPTsndDelta nciSndDelta SNVT_lux (79)	27	nv1, nv2, nv3, nv4, nv5, nv6, nv7, nv8	Sets the minimum brightness change required to force transmission of the output value

* 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.

7.2. Network Variables

7.2.1. Sun Sensor Output [x]

```
network output SNVT_lux nvoSunLux_[x];
```

This output network variable reports sun (outdoor) brightness with a range from 0…65335 lx.

Valid Range

0 … 65335 lx (resolution 1 lx)

Default Value

0 lx.(Lux)

Configuration Considerations

Behaviour depends on the values of the following properties:

SCPTmaxSendTime

SCPTminSendTime

SCPTsndDelta

7.3. Configuration Properties

7.3.1. Max Send Time

```
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 – nvoSunLux_1
- nv2 – nvoSunLux_2
- nv3 – nvoSunLux_3
- nv4 – nvoSunLux_4
- nv5 – nvoSunLux_5
- nv6 – nvoSunLux_6
- nv7 – nvoSunLux_7
- nv8 – nvoSunLux_8

Valid Range

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

Default Value

60.0 seconds.

SCPT Reference

SCPTmaxSendTime (49)

7.3.2. 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 – nvoSunLux_1
- nv2 – nvoSunLux_2
- nv3 – nvoSunLux_3
- nv4 – nvoSunLux_4
- nv5 – nvoSunLux_5
- nv6 – nvoSunLux_6
- nv7 – nvoSunLux_7
- nv8 – nvoSunLux_8

Valid Range

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

Default Value

10.0 seconds.

SCPT Reference

SCPTminSendTime (52)

7.3.3. Send On Delta Value

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

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

- nv1 – nvoSunLux_1
- nv2 – nvoSunLux_2
- nv3 – nvoSunLux_3
- nv4 – nvoSunLux_4
- nv5 – nvoSunLux_5
- nv6 – nvoSunLux_6
- nv7 – nvoSunLux_7
- nv8 – nvoSunLux_8

Valid Range

0 … 65335 lx.

Default Value

4000 lx.(Lux)..

SCPT Reference

SCPTsndDelta (27)

8. Wind Sensors #20002

8.1. Functional-Block Details

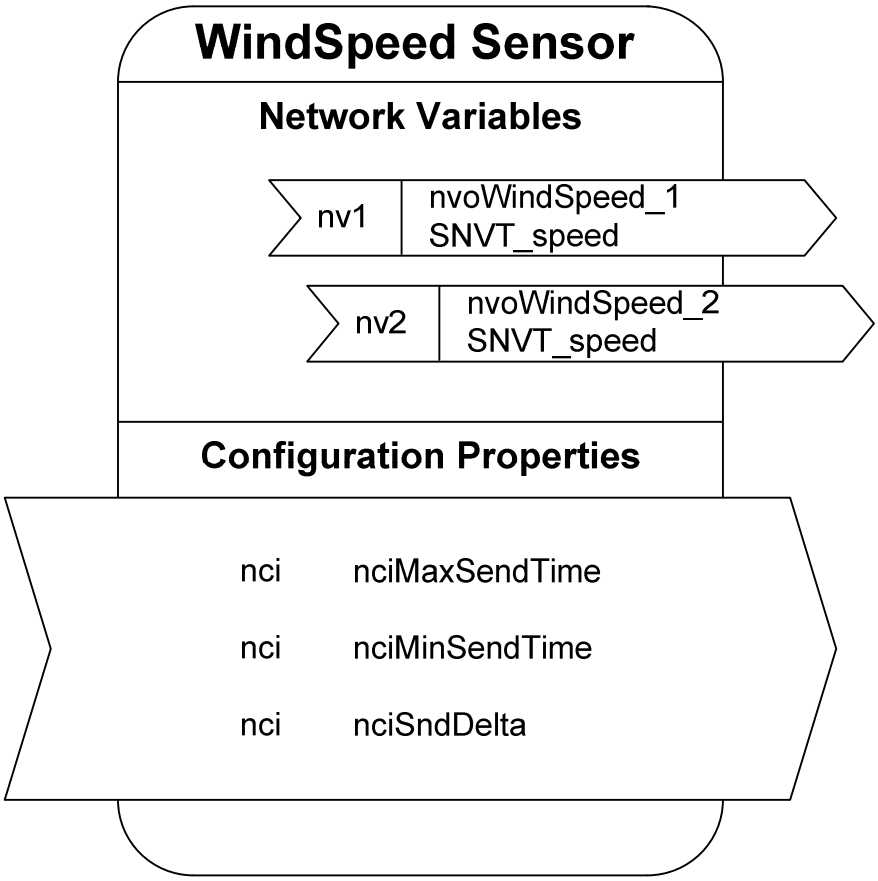


Table 1 SNVT Details

NV (M/O)*	Variable Name	SNVT Name	SNVT Index	Description
1(M)	nvoWindSpeed_1	SNVT_speed	34	Wind speed sensor output
2(M)	nvoWindSpeed_2	SNVT_speed	34	Wind speed sensor output

* M = mandatory, O = optional

Table 2 SCPT Details

(M/O) **	SCPT Name	SCPT/UCPT Index	Associated NVs**	Description
(M)	SCPTmaxSendTime nciMaxSendTime SNVT_time_sec (107)	49	nv1, nv2	Maximum period of time that expires before the object will automatically update NV
(M)	SCPTminSendTime nciMinSendTime SNVT_time_sec (107)	52	nv1, nv2	Minimum period of time that expires before the object will automatically update NV
(M)	UCPTsndDeltaWindSpeed nciSndDelta SNVT_speed (34)	#10	nv1, nv2	Sets the minimum speed change required to force transmission of the output value

* 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.

8.2. Network Variables

8.2.1. Wind Speed Output [x]

```
network output SNVT_speed  nvoWindSpeed_[x];
```

This output network variable reports wind speed with a range from 0…6533,5m/s.

Valid Range

0 … 6533.5 m/s (Resolution 0,1m/s)

Default Value

0 m/s.

Configuration Considerations

Behaviour depends on the values of the following properties:

SCPTmaxSendTime

SCPTminSendTime

SCPTsndDelta

8.3. Configuration Properties

8.3.1. Max Send Time

```
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 – nvoWindSpeed_1
- nv2 – nvoWindSpeed_2

Valid Range

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

Default Value

60.0 seconds.

SCPT Reference

SCPTmaxSendTime (49)

8.3.2. 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 – nvoWindSpeed_1
- nv2 – nvoWindSpeed_2

Valid Range

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

Default Value

10.0 seconds.

SCPT Reference

SCPTminSendTime (52)

8.3.3. Send On Delta Value

```
network input config SNVT_speed nciSndDelta;
```

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

- nv1 – nvoWindSpeed_1
- nv2 – nvoWindSpeed_2

Valid Range

0 … 6533,5 m/s (Resolution 1 m/s)

Default Value

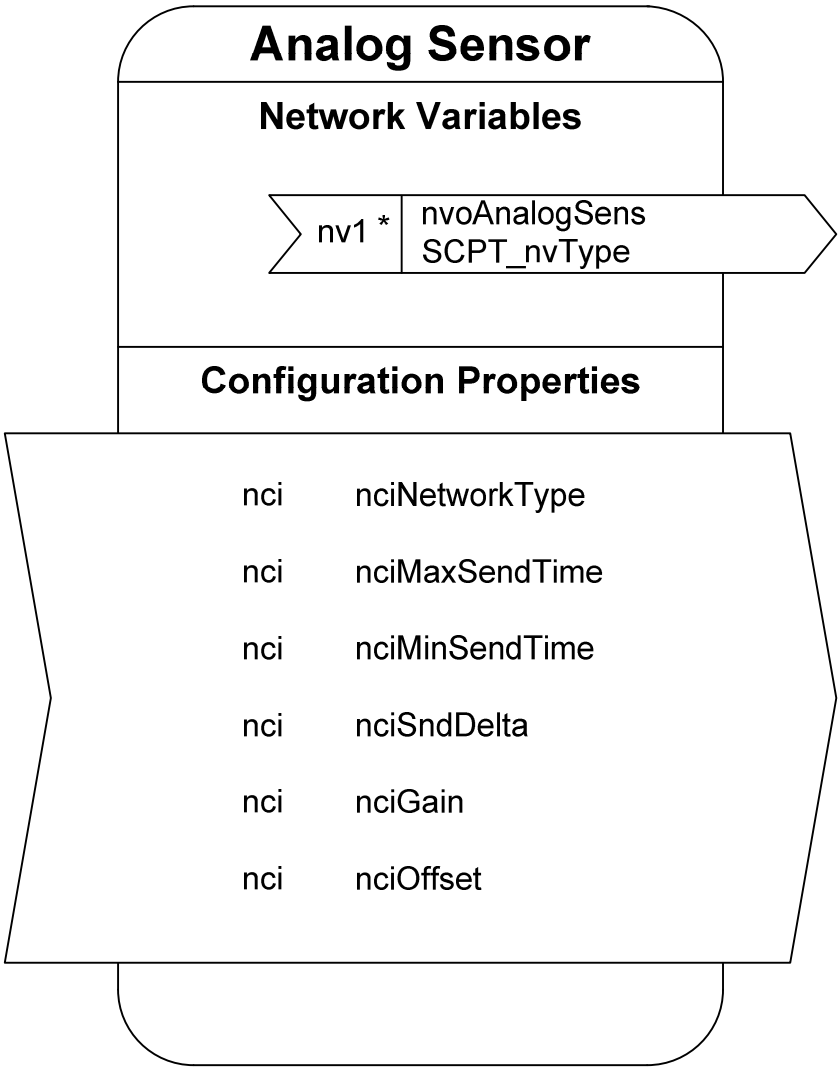
1,4 m/s.

SCPT Reference

SCPTsndDelta (27)

9. Analog Sensor #0001

9.1. Functional-Block Details



* The Functional Block “Analog Sensor” exists 3 times at the animeo LON Sensor Interface. Each could be set individually and separately from the others.

Table 1 SNVT Details

NV (M/O)*	Variable Name	SNVT Name	Description
1(M)	nvoAnalogSens_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 (Global Radiation, CO2, Temperature or Brightness)
(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.

9.2. Network Variables

9.2.1. Analog Sensor Output x

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

This output network variable reports the value of the sensor which is used (Global radiation,CO2, Temperature or Brightness).

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

9.3. Configuration Properties

9.3.1. Type Of Analog Sensor

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

Valid Range

nvoAnalogSensorOut	Meaning
SNVT_power	Global Radiation
SNVT_lev_percent	CO2
SNVT_temp_p	Temperature
SNVT_lux	Brightness

Default Value

3 = Temperature

SCPT Reference

SCPTnvType (254)

9.3.1.1. Global Radiation As Analog Sensor

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

This input configuration property sets the global radiation value into watt (W) at the following network variables:

- nv1 – nvoAnalogSensorOut_x

Valid Range of Analog Sensor Output

0,0 … 6553,5W (Resolution 0.1W)

Default Value of Analog Sensor Output

0 W (Watt)

9.3.1.2. CO2 As Analog Sensor

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

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

- nv1 – nvoAnalogSensorOut_x

Valid Range of Analog Sensor Output

-163.840 … 163.830 ppm (Resolution 0.001ppm)

Default Value of Analog Sensor Output

0 ppm (parts-per-million)

9.3.1.3. Temperature As Analog Sensor

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

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

- nv1 – nvoAnalogSensorOut_x

Valid Range of Analog Sensor Output

-273.17 … 327.66°C (Resolution 0.01°C)

Default Value of Analog Sensor Output

0 °C (Degree Celsius)

9.3.1.4. 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 – nvoAnalogSensorOut_x

Valid Range of Analog Sensor Output

0 … 65535 lx (Resolution 1.0 lx)

Default Value of Analog Sensor Output

0 lx

9.3.2. Max Send Time

```
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 – nvoAnalogSensorOut_x

Valid Range

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

Default Value

60.0 seconds.

SCPT Reference

SCPTmaxSendTime (49)

9.3.3. 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 – nvoAnalogSensorOut_x

Valid Range

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

Default Value

10.0 seconds.

SCPT Reference

SCPTminSendTime (52)

9.3.4. Send On Analog Sensor Delta

SCPT Reference

SCPTsndDelta (27)

9.3.4.1. Send On Power Delta

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

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

- nv1 – nvoAnalogSensorOut_x

Valid Range

0,0 … 6553,5W (Resolution 0.1W)

Default Value

10 W (Watt)

9.3.4.2. Send On Percent (CO2) Delta

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

This input configuration property sets the change in value that must result before the Object

will automatically (cyclically) update the following network variables:

- nv1 – nvoAnalogSensorOut_x

Valid Range

-163.840 … 163.830 ppm (Resolution 0.005ppm)

Default Value

500 ppm (parts-per-million)

9.3.4.3. Send On Temperatur Delta

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

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

- nv1 – nvoAnalogSensorOut_x

Valid Range

-273.17 … 327.66°C (Resolution 0.01°C)

Default Value

1 °C (Degree Celsius)

9.3.4.4. Send On Brightness Delta

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

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

- nv1 – nvoAnalogSensorOut_x

Valid Range

0.0 … 65535 lx (Resolution 1.0 lx)

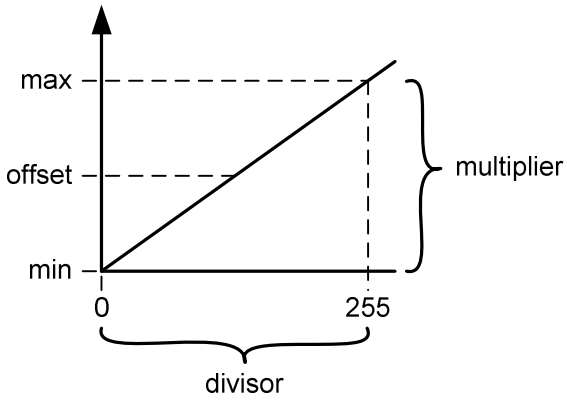
Default Value

10 lx (Lux)

9.3.5. 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:



9.3.6. Analog Sensor Offset

SCPT Reference

SCPToffset (26)

9.3.6.1. 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 – nvoAnalogSensorOut_x

Valid Range

0.0 … 6553.5 W (Resolution 0.1W)

Default Value

0 W (Watt)

9.3.6.2. 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 – nvoAnalogSensorOut_x

Valid Range

-163.840 … 163.830 ppm (Resolution 0.005 ppm)

Default Value

0 ppm (parts-per-million)

9.3.6.3. 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 – nvoAnalogSensorOut_x

Valid Range

-273.17 … 327.66°C (Resolution 0.01°C)

Default Value

0°C (Degree Celsius)

9.3.6.4. 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 – nvoAnalogSensorOut_x

Valid Range

0 … 65535 lx (Resolution 1.0 lx)

Default Value

0 lx