

Beschreibung der Software

LONSE

80:00:57:05:01:06:04:00(REVISION B)

Schnittstellenbeschreibung und Software-Dokumentation (Technische Änderungen vorbehalten)

Echelon, LON, Neuron, 3150, 3120 LONTALK und LONMARK sind Handelsmarken bzw. eingetragene Handelsmarken der Echelon Corporation. Andere Marken und Produktnamen sind Handelsmarken bzw. eingetragene Handelsmarken Anderer.

1	Allgemeines	5
1.1	Gerätebeschreibung.....	5
1.2	Typografie	5
1.3	Einschränkungen	5
1.4	Wink-Funktion	5
1.5	Objekte.....	6
2	Node Object #0	7
2.1	Schnittstellenbeschreibung	7
2.1.1	Eingang Object Request.....	7
2.1.2	Ausgang Object Status	7
2.1.3	Eingang Uhrzeit und Datum.....	7
2.1.4	Ausgang Alarm	7
3	Temperatursensor.....	8
3.1	Schnittstellenbeschreibung	8
3.1.1	Ausgang Temperatur	8
3.2	Configuration Properties	8
3.2.1	Send on Delta Wert	8
3.2.2	Temperatur Offset.....	9
3.2.3	Temperatur Gain.....	10
3.2.4	Sensortyp.....	10
4	Feuchtigkeitssensor	12
4.1	Schnittstellenbeschreibung	12
4.1.1	Ausgang Relative Luftfeuchtigkeit	12
4.2	Configuration Properties	12
4.2.1	Send on Delta Wert	12
4.2.2	Feuchtigkeit Offset.....	13
4.2.3	Feuchtigkeit Gain.....	13
4.2.4	Sensortyp.....	13
5	Windrichtungsgeber	14
5.1	Schnittstellenbeschreibung	14
5.1.1	Ausgang Windrichtung.....	14
5.2	Configuration Properties	15
5.2.1	Send on Delta Wert	15
5.2.2	Windrichtung Offset	15
5.2.3	Windrichtung Gain	15
5.2.4	Sensortyp.....	15
6	Helligkeitssensor Standard.....	16
6.1	Schnittstellenbeschreibung	16
6.1.1	Ausgang Helligkeit	16
6.2	Configuration Properties	17
6.2.1	Send on Delta Wert	17
6.2.2	Lux Offset.....	17
6.2.3	Lux Gain.....	17
7	Windgeschwindigkeitsgeber.....	18
7.1	Schnittstellenbeschreibung	18
7.1.1	Ausgang Windgeschwindigkeit	18
7.2	Configuration Properties	18
7.2.1	Send on Delta Wert	18
7.2.2	Windgeschwindigkeit Offset.....	18
7.2.3	Windgeschwindigkeit Gain.....	19
7.2.4	Sensortyp.....	19
8	Niederschlagsgeber	20
8.1	Schnittstellenbeschreibung	20
8.1.1	Ausgang Niederschlag.....	20
8.2	Configuration Properties	20
9	Real Time Keeper #1 (DCF77-Funkuhr).....	21

9.1	Allgemeines zum DCF77 Signal	21
9.2	Funktion	21
9.3	Inbetriebnahme	21
9.4	Schnittstellenbeschreibung	21
9.4.1	Ausgang Time Stamp	22
9.4.2	Ausgang DayDate.....	22
9.4.3	Ausgang MEZMESZ	23
9.4.4	Manuelles stellen der Uhr über das Node Object.....	23
9.5	Configuration Properties	24
9.5.1	Manuelle Einstellung MEZ/MESZ	24
10	Sonnenstandsberechnung #20002.....	25
10.1	Begriffe zur Sonnenstandsberechnung.....	25
10.2	Schnittstellenbeschreibung #20002	25
10.2.1	Ausgang Elevation.....	25
10.2.2	Ausgang Azimut.....	25
10.3	Configuration Properties	26
10.3.1	Längengrad.....	26
10.3.2	Breitengrad	26
10.3.3	Differenz zur MEZ.....	27
11	Helligkeitssensor Astro #20001	29
11.1	Schnittstellenbeschreibung	29
11.1.1	Ausgang Helligkeit / Astro.....	29
11.2	Configuration Properties	30
12	Eisüberwachung.....	31
12.1	Allgemeines zur Eisüberwachung.....	31
12.2	Funktion	31
12.3	Schnittstellenbeschreibung	31
12.3.1	Ausgang Eisüberwachung	31
12.4	Configuration Properties	32
12.4.1	Schaltsschwelle Eisüberwachung	32
13	REKO Object #20000.....	33
13.1	Schnittstellenbeschreibung	33
13.1.1	Eingang Sendezzeiten Netzwerkvariablenverwaltung	33
13.1.2	Ausgang Sendezzeiten Netzwerkvariablenverwaltung	35
13.1.3	Ausgang Funktionstest	36
14	Quellenangaben	37

1 Allgemeines

1.1 Gerätebeschreibung

Das LONWORKS-Gerät LONSE hat folgende Funktionalität:

- Erfassung verschiedener Meteorologischer Meßgrößen
- Echtzeituhr über die Erfassung des DCF77 Funkuhrsignals
- Sonnenstandsberechnung

Es werden folgende Meteorologischen Meßgrößen erfaßt:

- 1 x Temperatur
- 1 x relative Luftfeuchtigkeit
- 8 x Helligkeit
- 3 x Windgeschwindigkeit
- 1 x Windrichtung
- 1 x Niederschlag
- 1 x Eiswarnung

Die Echtzeituhr stellt die Uhrzeit sowie das Datum zur Verfügung, zusätzlich wird noch der Wochentag bereitgestellt.

Die Sonnenstandsberechnung liefert Elevation und Azimut der Sonne.

Das Gerät ist von LONMARK® zertifiziert.

Technische Daten sowie Informationen über den Anschluß des LONSE sind der „Benutzer- und Installationsanleitung Sensoreinheit LONSE“ [5] zu entnehmen.

1.2 Typografie

- Standardtext in Arial Standard 10 pt
- *Dokumentenverweise in Arial Kursiv 10 pt*
- Auszüge aus NEURON-C-Quelltexten in Courier New Standard 10 pt
- Darstellung von Dezimalzahlen 12345
- Darstellung von Hexadezimalzahlen 0x05

1.3 Einschränkungen

Soweit nicht anders beschrieben, sind alle hier aufgeführten Datenformate und Funktionalitäten entsprechend LONMARK-Konventionen bzw. nach [1] zu verstehen.

1.4 Wink-Funktion

Wird auf dem Knoten die Wink-Funktion ausgeführt leuchtet die Service-LED für ca. 1 Sekunde auf und erlischt anschließend wieder.

1.5 Objekte

Ein Objekt stellt eine geschlossene abstrakte Einheit aus Verhalten (Funktionalität) und Eigenschaften (Daten) dar; es kann Schnittstellen zur Kommunikation mit der Außenwelt (z.B. andere Objekte) zur Verfügung stellen. Als Basis der hier definierten Objekte dienen die generischen Objekte, wie sie in [2] beschrieben sind. In der Applikation sind mehrere Objekte implementiert.

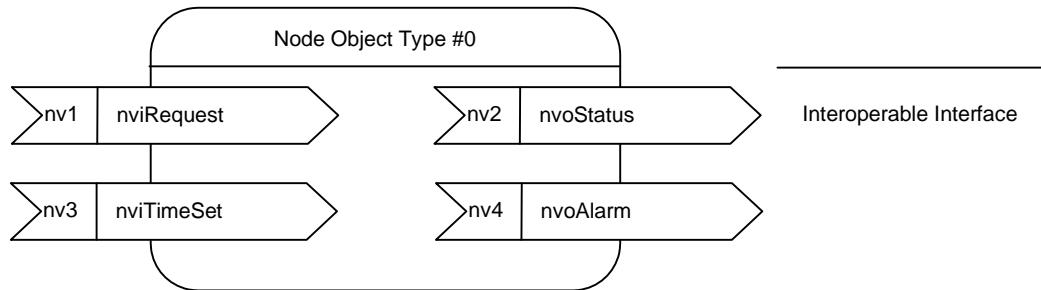
Alle genannten Objekte sind gleichzeitig auf einem Gerät implementiert.

Objekt Nr.	Objekt Typ	Funktion
0	#0 Node Object	Knotenverwaltung
1	#1 OLS (Open Loop Sensor Object)	Temperatur
2	#1 OLS	Relative Luftfeuchtigkeit
3	#1 OLS	Windrichtung
4-11	#1 OLS	Helligkeit
12-14	#1 OLS	Windgeschwindigkeit
15	#1 OLS	Niederschlag
16	#1 OLS	Funkuhr
17	#20002	Sonnenstands Berechnung
18-25	#20001	Helligkeit/Sonnenstand
26	#1 OLS	Eisüberwachung
27	#20000	Sendezeiten/Test

Tabelle 1 Übersicht der vorhandenen Objekte

2 Node Object #0

2.1 Schnittstellenbeschreibung



2.1.1 Eingang Object Request

```
network input SNVT_obj_request nviRequest
```

Unterstützte Dienste:

RQ_NORMAL	Setzt „disabled“ Objekt „enabled“ , wenn dieser Dienst für das Node Objekt ausgewählt wird werden alle Objekte des Knoten „enabled“ und gegebenenfalls ein Selbsttest beendet
RQ_DISABLED	Setzt „enabled“ Objekt „disabled“, wenn dieser Dienst für das Node Objekt ausgewählt wird werden alle Objekte des Knoten „disabled“
RQ_REPORT_MASK	Die Status Bits der unterstützten Dienste werden an nvoStatus gemeldet
RQ_CLEAR_STATUS	Setzt alle Status Bits in nvoStatus zurück
RQ_UPDATE_STATUS	Der Status des ausgewählten Objektes wird in nvoStatus ausgegeben
RQ_SELF_TEST	Nur für Real Time Keeper (Objekt 16). Führt einen Selbsttest für das Objekt durch (Siehe auch 9.3)

2.1.2 Ausgang Object Status

```
network output SNVT_obj_status nvoStatus;
```

Stellt Statusbits des Knotens zur Verfügung [1].

2.1.3 Eingang Uhrzeit und Datum

```
network input SNVT_time_stamp nviTimeSet;
```

Dient zum Stellen der internen Uhr. Ist ein Funkuhrempfänger angeschlossen und wird das Funkuhrsignal erfolgreich empfangen synchronisiert sich die interne Uhr selbständig auf die Funkuhrzeit.

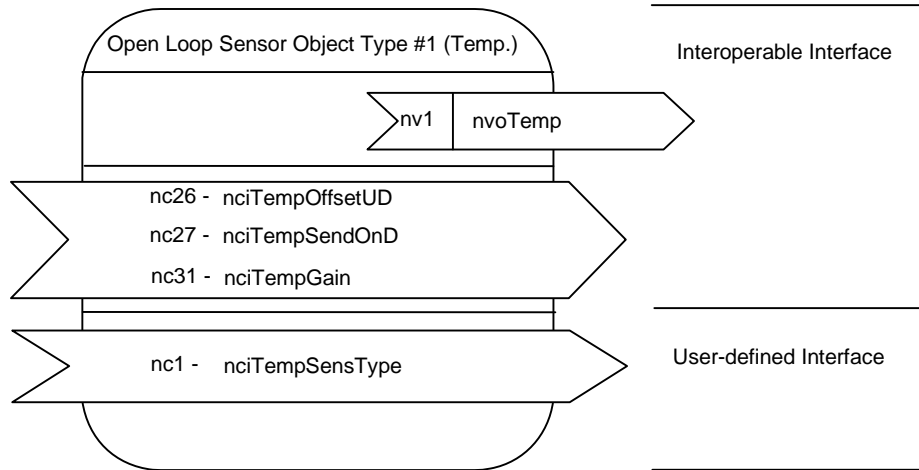
2.1.4 Ausgang Alarm

```
network output SNVT_alarm nvoAlarm;
```

Wird momentan noch nicht ausgewertet.

3 Temperatursensor

3.1 Schnittstellenbeschreibung



3.1.1 Ausgang Temperatur

network output SNVT_temp_p nvoTemp;

Diese Variable stellt auf dem Netzwerk einen Temperaturwert zur Verfügung

Valid Range

nvoTemp	-273,17..327,66°C (0,01°C)
---------	----------------------------

Default Value

Nach Reset des Gerätes hat nvoTemp den Wert:

nvoTemp	0x7FFF
---------	--------

Auflösung

Auflösung bei Einsatz eines Sensors mit einem Meßbereich von 100 °C

nvoTemp	0,39 °C
---------	---------

When Transmitted

Die MaxSendTime und MinSendTime für die Netzwerkvariable nvoTemp wird über die nviConfigTime (Siehe 13.1.1) parametrisiert.

3.2 Configuration Properties

3.2.1 Send on Delta Wert

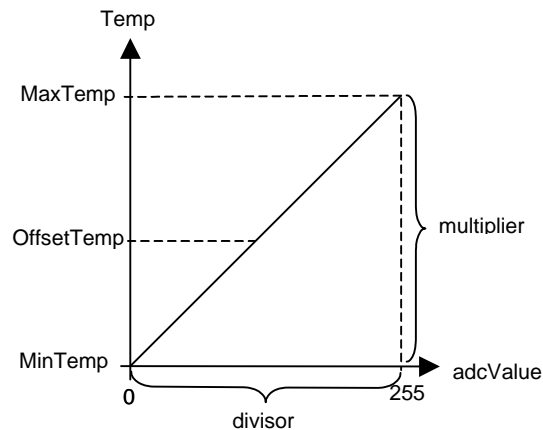
network input config SNVT_temp_p nciTempSendOnD;

Diese Variable legt fest um welchen Betrag sich die Variable ändern muß bevor sie nach Ablauf von MinSendTime gesendet wird.

3.2.2 Temperatur Offset

```
network input config SNVT_temp_p nciTempOffset;
```

Die Konfigurationsvariable `nciTempOffset` ermöglicht in Verbindung mit `nciTempGain` den Anschluß benutzerdefinierter Sensoren mit linearer Kennlinie. Voraussetzung ist, dass der Sensor über entsprechende elektrische Eigenschaften verfügt (0-10V, siehe auch [5]).



3.2.3 Temperatur Gain

```
network input config SNVT_muldiv nciTempGain;
```

Diese Variable legt einen Verstärkungsfaktor für die an `nvoTemp` zur Verfügung stehenden Werte fest. Die hier parametrisierten Werte werden nur verwendet wenn als Sensortyp „Userdefined“ parametrisiert wird.

Element in	Bedeutung
<code>nciTempGain</code>	
<code>.unsigned long</code>	multiplier
<code>.unsigned long</code>	divisor

Beispiel:

Ein Temperatursensor mit einem Meßbereich von -25°C bis $+75^{\circ}\text{C}$ soll verwendet werden.
 $-25^{\circ}\text{C} \dots +75^{\circ}\text{C} \Rightarrow$ Meßbereich von 100°C , die Temperatur wird in $0,01^{\circ}$ Auflösung angegeben \Rightarrow multiplier = 10000.

Die Auflösung des A/D Wandlers beträgt 255 Schritte \Rightarrow divisor = 255 (Diese Einstellung sollte sinnvollerweise nicht verändert werden).

Der Meßbereich des Sensors beginnt bei $-25^{\circ}\text{C} \Rightarrow$ `nciTempOffset` = -2500

3.2.4 Sensortyp

```
network input config UCPTsensType nciTempSensType;
```

Über die Konfigurationsvariable `nciTempSensType` kann zwischen vorparametrisierten Sensoren und einem selbst definierten Sensor gewählt werden. Wird der selbst definierte Sensor gewählt, müssen auch die Variablen `nciTempGain` und `nciTempOffset` entsprechend parametrisiert werden. Wird ein vorparametrisierter Sensor gewählt, müssen diese drei Konfigurationsvariablen nicht parametrisiert werden (Resource files: `UCPTsensType`).

Valid Range

<code>nciTempSensType</code>	0	<code>SensorTypeUser</code>	
	1	<code>SensorType1</code>	Art.Nr. : 628029
	2	<code>SensorType2</code>	Art.Nr. : 628146

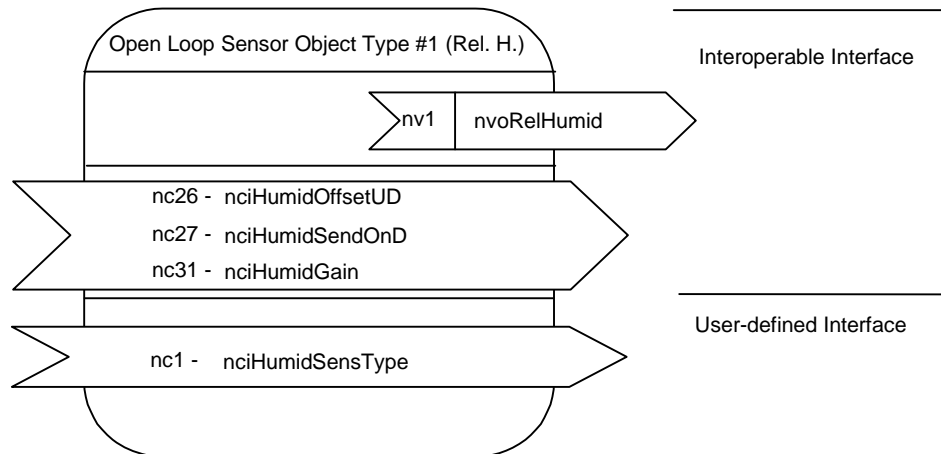
Default Value

Nach Inbetriebnahme des Gerätes hat nciTempSensType den Wert:

nciTempSensType	1	SensorTyp1	Art.Nr. : 628029
-----------------	---	------------	------------------

4 Feuchtigkeitssensor

4.1 Schnittstellenbeschreibung



4.1.1 Ausgang Relative Luftfeuchtigkeit

network output SNVT_lev_cont nvoRelHumid;

Diese Variable stellt auf dem Netzwerk einen relativen Luftfeuchtigkeitswert zur Verfügung

Valid Range

nvoRelHumid	0..100% (0,5%)
-------------	----------------

Default Value

Nach Reset des Gerätes hat nvoRelHumid den Wert:

nvoRelHumid	0xFF
-------------	------

Auflösung

Auflösung bei Einsatz eines Sensors mit einem Meßbereich von 0..100 %

nvoRelHumid	0,5%
-------------	------

When Transmitted

Die MaxSendTime und MinSendTime für die Netzwerkvariable nvoRelHumid wird über die nviConfigTime (Siehe 13.1.1) parametrisiert.

4.2 Configuration Properties

4.2.1 Send on Delta Wert

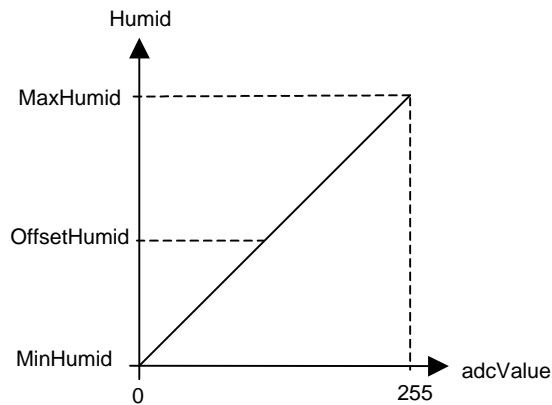
network input config SNVT_lev_cont nciHumidSendOnD;

Diese Variable legt fest um welchen Betrag sich die Variable ändern muß, bevor sie nach Ablauf von MinSendTime gesendet wird.

4.2.2 Feuchtigkeit Offset

```
network input config SNVT_lev_cont nciHumidOffset;
```

Die Konfigurationsvariable `nciHumidOffset` ermöglicht in Verbindung mit `nciHumidGain` den Anschluß benutzerdefinierter Sensoren mit linearer Kennlinie. Voraussetzung ist, dass der Sensor über entsprechende elektrische Eigenschaften verfügt (0-10V, siehe auch [5])!



4.2.3 Feuchtigkeit Gain

```
network input config SNVT_muldiv nciHumidGain;
```

Diese Variable legt einen Verstärkungsfaktor für die an `nvoRelHumid` zur Verfügung stehenden Werte fest (Siehe 3.2.3).

4.2.4 Sensortyp

```
network input config UCPTsensType nciHumidSensType;
```

Über die Konfigurationsvariable `nciHumidSensType` kann zwischen vorparametrierten Sensoren und einem selbst definierten Sensor gewählt werden. Wird der selbst definierte Sensor gewählt müssen auch die Variablen `nciHumidGain` und `nciHumidOffset` entsprechend parametrieren werden. Wird ein vorparametriertes Sensor gewählt müssen diese drei Konfigurationsvariablen nicht parametrieren werden (Resource files: `UCPTsensType`).

Valid Range

<code>nciHumidSensType</code>	0	<code>SensorTypeUser</code>	
	1	<code>SensorType1</code>	Art.Nr.: 623047
	2	<code>SensorType2</code>	Art.Nr.: 628147

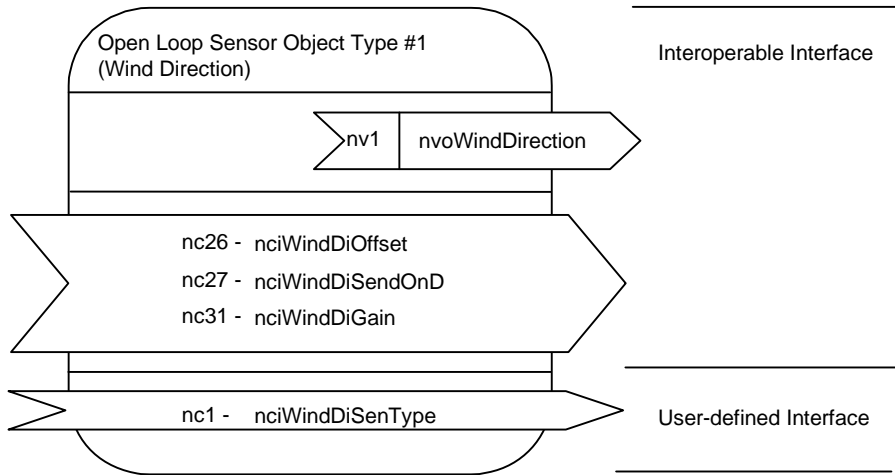
Default Value

Nach Inbetriebnahme des Gerätes hat `nciHumidSensType` den Wert:

<code>nciHumidSensType</code>	1	<code>SensorType1</code>	Art.Nr.: 623047
-------------------------------	---	--------------------------	-----------------

5 Windrichtungsgeber

5.1 Schnittstellenbeschreibung



5.1.1 Ausgang Windrichtung

```
network output SNVT_angle_deg nvoWindDirection;
```

Hier wird ein 10-min-Mittel aus Einzelwerten berechnet. Es wird nur eine gültige Windrichtung ausgegeben wenn die Windgeschwindigkeit des 1.ten Windgeschwindigkeitsgebers innerhalb der letzten Minute einmal größer 0,5 m/s war. Andernfalls wird als Windrichtung 0° ausgegeben. Daher muß der erste Windrichtungsgeber in unmittelbarer Nähe zum 1.ten Windgeschwindigkeitsgebers montiert werden, siehe auch [5].

Der erste gültige Windrichtungswert wird frühestens 10 Minuten nach Reset des Gerätes ausgegeben.

Valid Range

nvoWindDirection	-359,98..360°
------------------	---------------

Default Value

Nach Reset des Gerätes hat nvoWindDirection den Wert:

nvoWindDirection	0x7FFF
------------------	--------

Auflösung

Auflösung bei Einsatz eines Sensors mit einem Meßbereich von 0..360 °

nvoWindDirection	1,5°
------------------	------

When Transmitted

Die MaxSendTime und MinSendTime für die Netzwerkvariable nvoWindDirection wird über die nviConfigTime (Siehe 13.1.1) parametrisiert.

5.2 Configuration Properties

5.2.1 Send on Delta Wert

```
network input config SNVT_angle_deg nciWindDiSendOnD;
```

Diese Variable legt fest um welchen Betrag sich die Variable ändern muß, bevor sie nach Ablauf von MinSendTime gesendet wird.

5.2.2 Windrichtung Offset

```
network input config SNVT_angle_deg nciWindDiOffset;
```

Die Funktionalität dieser Konfigurationsvariable ist noch nicht implementiert!

5.2.3 Windrichtung Gain

```
network input config SNVT_muldiv nciWindDiGain;
```

Die Funktionalität dieser Konfigurationsvariable ist noch nicht implementiert!

5.2.4 Sensortyp

```
network input config UCPTsensType nciWindSpSenType[X];
```

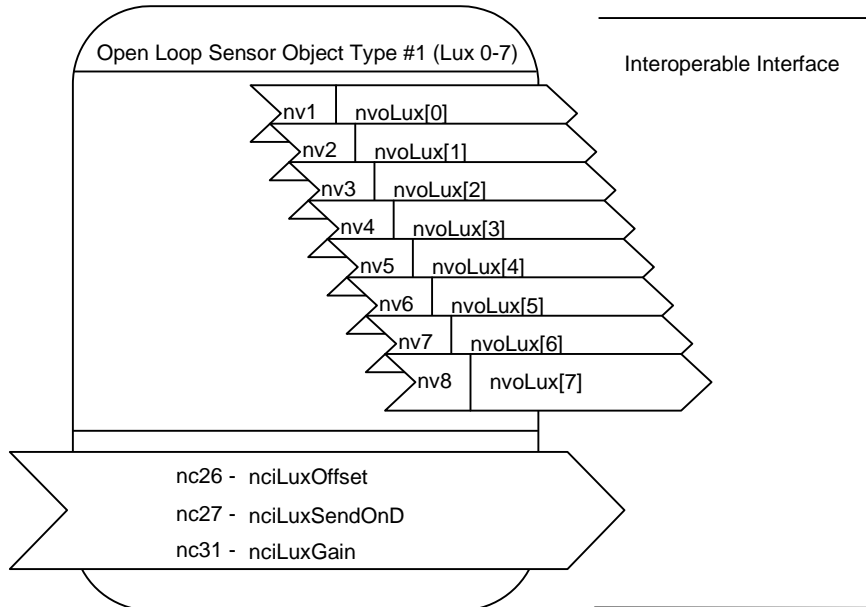
Die Funktionalität dieser Konfigurationsvariable ist noch nicht implementiert!

Valid Range

Default Value

6 Helligkeitssensor Standard

6.1 Schnittstellenbeschreibung



6.1.1 Ausgang Helligkeit

network output SNVT_Lux nvoLux[X];

Diese Variable stellt auf dem Netzwerk einen Helligkeitswert zur Verfügung.

Valid Range

nvoLux[0]	0..65535 lux (1 lux)
-----------	----------------------

Default Value

Nach Reset des Gerätes hat nvoLux[X] den Wert:

nvoLux[X]	0xFFFF
-----------	--------

Auflösung

Auflösung bei Einsatz eines Sensors mit einem Meßbereich von 100000 lux

nvoLux[X]	392 lux
-----------	---------

When Transmitted

Die MaxSendTime und MinSendTime für die Netzwerkvariable nvoLux[X] wird über die nviConfigTime (Siehe 13.1.1) parametrier. Alle sechs Helligkeitsvariablen werden über einen Parametersatz parametrier!

6.2 Configuration Properties

6.2.1 Send on Delta Wert

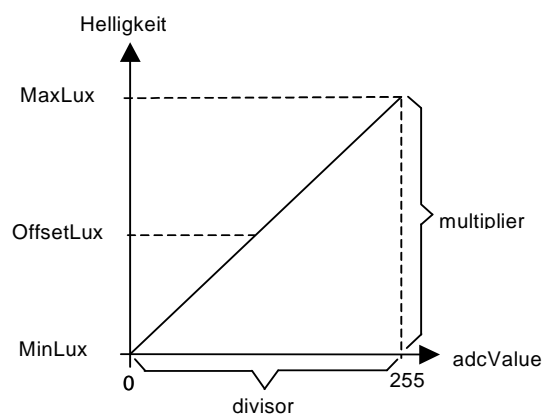
```
network input config SNVT_lux nciLuxSendOnD;
```

Diese Variable legt fest um welchen Betrag sich die Variable ändern muß, bevor sie nach Ablauf von MinSendTime gesendet wird.

6.2.2 Lux Offset

```
network input config SNVT_lux nciLuxOffset;
```

Die Konfigurationsvariable `nciLuxOffset` ermöglicht in Verbindung mit `nciLuxGain` den Anschluß benutzerdefinierter Sensoren mit linearer Kennlinie. Voraussetzung ist, dass der Sensor über entsprechende elektrische Eigenschaften verfügt (0-10V, 0-20mA Ausgang, siehe auch [5]).



6.2.3 Lux Gain

```
network input config SNVT_muldiv nciLuxGain;
```

Diese Variable legt einen Verstärkungsfaktor für die an `nvoLux[X]` zur Verfügung stehenden Werte fest.

Element in	Bedeutung
<code>nciLuxGain</code>	
<code>.unsigned long</code>	multiplier
<code>.unsigned long</code>	divisor

Beispiel:

Ein Helligkeitssensor mit einem Meßbereich von 0 Lux bis 100000 Lux soll verwendet werden.
0 Lux ... 100000 Lux \Rightarrow Meßbereich von 100000 Lux, die Helligkeit wird in 1 Lux Auflösung angegeben \Rightarrow multiplier = 100000.

Da aber der Wert für multiplier maximal 65535 sein kann ist es nicht möglich 100000 zu parametrieren. Daher muß für die Variable ein anderer Wert gefunden werden, wobei das Verhältnis multiplier zu divisor gleich bleiben muß, z.B.:

$$\frac{\text{multiplier}}{\text{divisor}} = \frac{100000}{255} \approx \frac{65535}{167}$$

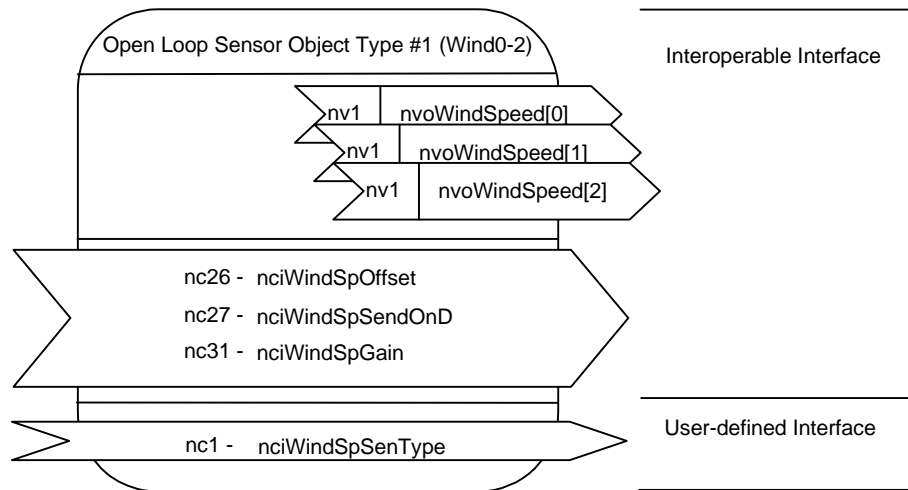
$$\text{multiplier} \rightarrow 65535$$

$$\text{divisor} \Rightarrow 167$$

Der Meßbereich des Sensors beginnt bei 0 Lux \Rightarrow `nciLuxOffset` = 0.

7 Windgeschwindigkeitsgeber

7.1 Schnittstellenbeschreibung



7.1.1 Ausgang Windgeschwindigkeit

```
network output SNVT_speed nvoWindSpeed[X];
```

Diese Variable stellt auf dem Netzwerk die Windgeschwindigkeit zur Verfügung

Valid Range

nvoWindSpeed[X]	0..65535,5 m/s (0,1m/s)
-----------------	-------------------------

Default Value

Nach Reset des Gerätes hat nvoWindSpeed[X] den Wert:

nvoWindSpeed[X]	0xFFFF
-----------------	--------

When Transmitted

Die MaxSendTime und MinSendTime für die Netzwerkvariable nvoWindSpeed[X] wird über die nviConfigTime (Siehe 13.1.1) parametrier. Alle drei Windgeschwindigkeitsvariablen werden über einen Parametersatz parametrier!

7.2 Configuration Properties

7.2.1 Send on Delta Wert

```
network input config SNVT_speed nciWindSpSendOnD;
```

Diese Variable legt fest um welchen Betrag sich die Variable ändern muß, bevor sie nach Ablauf von MinSendTime gesendet wird.

7.2.2 Windgeschwindigkeit Offset

```
network input config SNVT_speed nciWindSpOffset;
```

Die Funktionalität dieser Konfigurationsvariable ist noch nicht implementiert!

7.2.3 Windgeschwindigkeit Gain

```
network input config SNVT_muldiv nciWindSpGain;
```

Die Funktionalität dieser Konfigurationsvariable ist noch nicht implementiert!

7.2.4 Sensortyp

Über die Konfigurationsvariable `nciWindSpSenType` kann zwischen vorparametrierten Sensoren und einem selbst definierten Sensor gewählt werden (Resource files: UCPTsensType).

```
network input config UCPTsensType nciWindSpSenType[X];
```

Über die Konfigurationsvariable `nciWindSpSenType[X]` kann zwischen drei verschiedenen Windgeschwindigkeitssensoren gewählt werden.

Valid Range

nciWindSpSenType	0	SensorUserType	
	1	SensorType1	Art.Nr.: 618003
	2	SensorType2	Art.Nr.: 623042

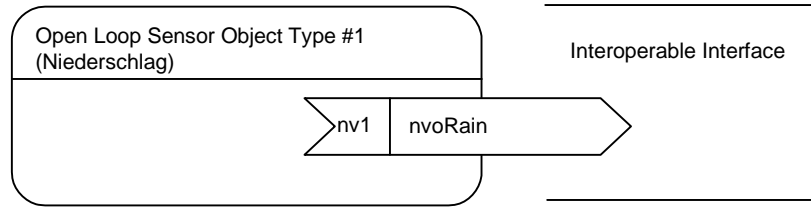
Default Value

Nach Inbetriebnahme des Gerätes hat `nciWindSpSenType[X]` den Wert:

nciWindSpSenType	1	SensorType1	Art.Nr.: 618003
------------------	---	-------------	-----------------

8 Niederschlagsgeber

8.1 Schnittstellenbeschreibung



8.1.1 Ausgang Niederschlag

```
network output SNVT_switch nvoRain;
```

Diese Variable stellt auf dem Netzwerk eine Information über Niederschlag zur Verfügung.

Valid Range

Element in	Kein Niederschlag	Niederschlag
nvoRain		
.value	0x00	0xC8
.state	0x00	0x01

Default Value

Nach Reset des Gerätes hat `nvoRelHumid` den Wert:

Element in	
nvoRain	
.value	0xFF
.state	0xFF

When Transmitted

Die `MaxSendTime` und `MinSendTime` für die Netzwerkvariable `nvoRain` wird über die `nviConfigTime` (Siehe 13.1.1) parametrisiert.

8.2 Configuration Properties

Keine.

9 Real Time Keeper #1 (DCF77-Funkuhr)

Dieses Objekt realisiert eine Echtzeituhr über die Auswertung des DCF77 Funkuhrsignals.

9.1 Allgemeines zum DCF77 Signal

Die Physikalisch Technische Bundesanstalt (PTB) sendet codierte Zeitsignale über den Langwellensender Mainflingen bei Frankfurt aus. Dieses Signal empfängt der Funkuhrempfänger, gibt es dem LONSE weiter und garantiert somit eine optimale Zeitgenauigkeit.

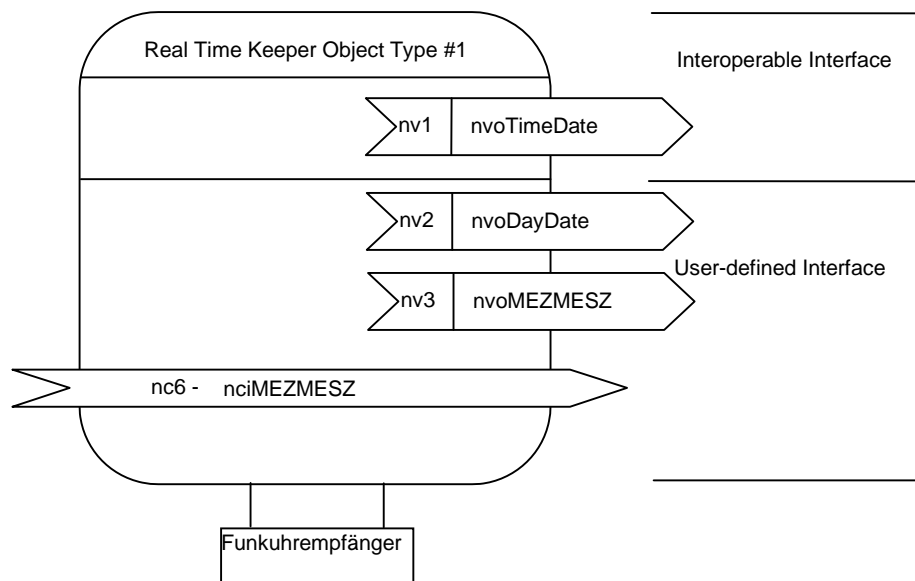
9.2 Funktion

Während jeder Minute werden die Werte von Minute, Stunde, Tag, Wochentag, Monat, Jahr, MEZ/MESZ, in codierten Zeitsignalen von der PTB übertragen. Dieses Funksignal wandelt der Funkuhrempfänger in elektrische Signale um, welche der LONSE entsprechend übernimmt. Die Atomuhr (Sender) stellt selbständig auf Sommer- / Winterzeit um, so daß ein manuelles Umstellen entfällt. Nach der Inbetriebnahme bzw. einem Stromausfall werden die aktuellen Zeitdaten innerhalb ca. 3 Minuten eingelesen und stehen dann bereit.

9.3 Inbetriebnahme

Für die Inbetriebnahme kann für das Objekt ein Selbsttest durchgeführt werden (Siehe 2.1.1). Während des Selbsttest wird nach dem Empfang eines Minutensynchronimpulses die Interne Uhr gestartet. Dies kann auch anhand der Netzwerkvariable `nvoTimeDate` beobachtet werden wenn die Sendezeit entsprechend parametrisiert ist (`MaxSendTime = 10`). Nach ca. 3-5 Minuten sollte die Zeitinformation korrekt eingelesen worden sein. Ist dies nach mehrmaligem Versuch nicht der Fall, ist der Montageort bzw. die Ausrichtung des DCF77 Empfängers zu ändern. Der Empfang des Funkuhrsignals kann auch anhand der Netzwerkvariablen `nvoFuncTest` überprüft werden (Siehe 13.1.3).

9.4 Schnittstellenbeschreibung



9.4.1 Ausgang Time Stamp

network output SNVT_time_stamp nvoTimeDate;

Die nvoTimeDate stellt anderen Knoten die Uhrzeit zur Verfügung.

Valid Range

Element in	
nvoTimeDate	
.year	0..3000
.month	0..12
.day	0..31
.hour	0..23
.minute	0..59
.second	0..59

Default Value

Nach Reset des Gerätes hat nvoTimeDate die Werte:

Element in	
nvoTimeDate	
.year	0xFFFF
.month	0xFF
.day	0xFF
.hour	0xFF
.minute	0xFF
.second	0xFF

When Transmitted

Die MaxSendTime für die Netzwerkvariable nvoTimeDate wird über die nviConfigTime (Siehe 13.1.1) parametrisiert.

9.4.2 Ausgang DayDate

network output SNVT_date_day nvoDayDate;

Die nvoDayDate stellt anderen Knoten den aktuellen Wochentag zur Verfügung.

Valid Range

nvoDayDate	Bedeutung
0 Day_SUN	Sonntag
1 Day_MON	Montag
2 Day_TUE	Dienstag
3 Day_WED	Mittwoch
4 Day_THU	Donnerstag
5 Day_FRI	Freitag
6 Day_SAT	Samstag
0xFF Day_NUL	ungültig

Default Value

Nach Reset des Gerätes hat nvoDateDay die Werte:

nvoDayDate	0xFF Day_NUL
------------	--------------

Wird das Funkuhrsignal erfolgreich eingelesen, wird der Wochentag aus diesem Signal generiert. Wird kein Funkuhrsignal empfangen, wird der Wochentag aus dem Datum berechnet.

When Transmitted

Die MaxSendTime für die Netzwerkvariable `nvoDayDate` wird über die `nviConfigTime` (Siehe 13.1.1) parametrisiert.

9.4.3 Ausgang MEZMESZ

```
network output SNVT_switch nvoMEZMESZ;
```

Die `nvoMEZMESZ` stellt anderen Knoten eine Information über die Sommer- bzw. Winterzeit zur Verfügung.

Valid Range

Element in	MEZ	MESZ
<code>nvoMEZMESZ</code>		
<code>.value</code>	0x00	0xC8
<code>.state</code>	0x00	0x01

Default Value

Nach Reset des Gerätes hat `nvoMEZMESZ` den Wert:

Element in	
<code>nvoMEZMESZ</code>	
<code>.value</code>	0xFF
<code>.state</code>	0xFF

Wird das Funkuhrsignal erfolgreich eingelesen, so wird MEZ/MESZ aus diesem Signal generiert. Wird kein Funkuhrsignal empfangen, so wird der über `nciMEZMESZ` parametrisierte Wert gesendet.

When Transmitted

Die MaxSendTime und MinSendTime für die Netzwerkvariable `nvoMEZMESZ` wird über die `nviConfigTime` (Siehe 13.1.1) parametrisiert.

9.4.4 Manuelles stellen der Uhr über das Node Object

```
network input SNVT_time_stamp nviTimeSet;
```

Mit dieser Netzwerkvariable ist es möglich die interne Uhr des LONSE ohne Funkuhrempfang zu stellen. Die Zeit `nvoTimeDate` beginnt dann ab dieser eingestellten Zeit zu laufen (Es muß eine Zeit eingestellt werden, die sich von den Default Werten unterscheidet!). Wird das Funkuhrsignal erfolgreich eingelesen, so wird `nvoTimeDate` automatisch auf die erfolgreich eingelesene Zeitinformation synchronisiert. Ist kein Funkuhrbetrieb erwünscht sollte der Funkuhrempfänger entfernt werden um ein unbeabsichtigtes Synchronisieren zu vermeiden (z.B. für Testzwecke oder bei Betrieb mit externer Systemuhr).

Valid Range

Element in	
<code>nviTimeSet</code>	
<code>.year</code>	0..3000
<code>.month</code>	0..12
<code>.day</code>	0..31
<code>.hour</code>	0..23
<code>.minute</code>	0..59
<code>.second</code>	0..59

Default Value

Element in nviTimeSet	
.year	1999
.month	1
.day	15
.hour	0
.minute	0
.second	0

Ist kein Funkuhrempfang möglich muß ein externes Uhrmodul auf den LONSE gebunden werden (Siehe Abbildung 1) um die Sonnenstandsberechnung nutzen zu können.

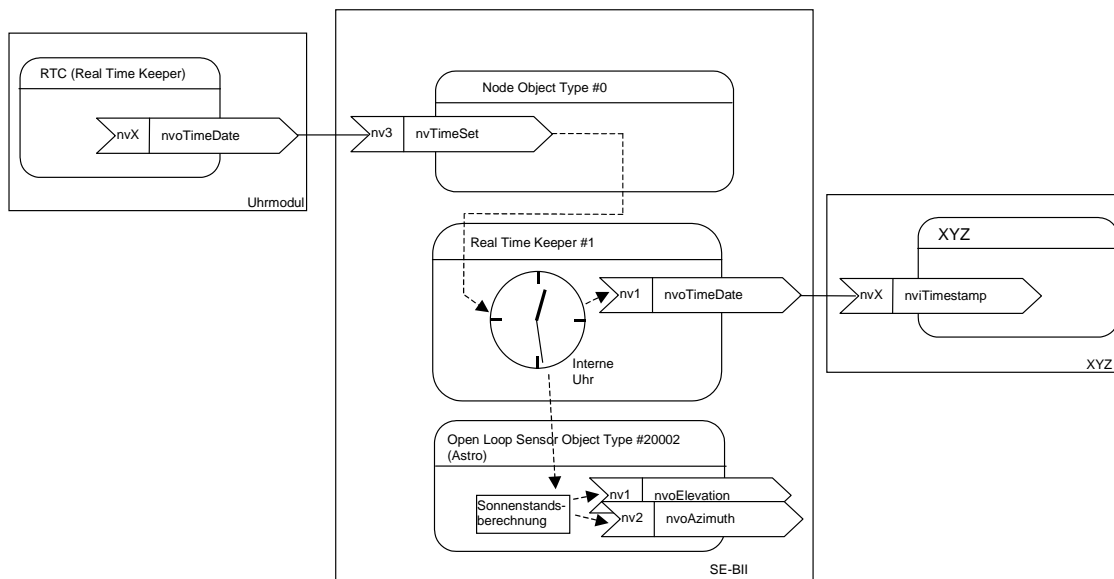


Abbildung 1, Betrieb mit externer Uhr

9.5 Configuration Properties

9.5.1 Manuelle Einstellung MEZ/MESZ

```
network input config SNVT_switch nciMEZMESZ;
```

Über diese Variable kann parametrisiert werden, ob eine über `nviTimeSet` empfangene Uhrzeit eine MEZ oder MESZ ist. Die entsprechende Information kann dann über `nvoMEZMESZ` anderen Knoten zur Verfügung gestellt werden. Die Information über die Zeitzone fließt auch in die Sonnenstandsberechnung ein! Ist ein Funkuhrempfänger angeschlossen und wird ein Funkuhrsignal erfolgreich eingelesen, so hat die in diesem Signal enthaltene Information über die Zeitzone höhere Priorität.

10 Sonnenstandsberechnung #20002

10.1 Begriffe zur Sonnenstandsberechnung

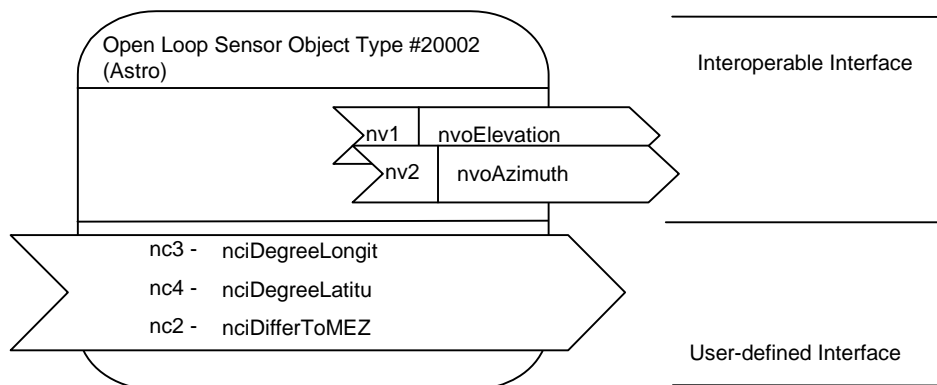
Sonnenhöhe (Elevation): Winkel zwischen dem Sonnenmittelpunkt und dem Horizont, vom Beobachter aus betrachtet; abhängig von Tageszeit, Jahreszeit, geographischer Länge und geographischer Breite des betreffenden Ortes.

Sonnenazimut (Azimut): Winkel zwischen der geographischen Nordrichtung und dem Vertikalkreis durch den Sonnenmittelpunkt (0° bis 360°); abhängig von Tageszeit, Jahreszeit, geographischer Länge und geographischer Breite des betreffenden Ortes.

Geographische Länge: Geographische Länge des Ortes für den der Sonnenstand berechnet werden soll.

Geographische Breite: Geographische Breite des Ortes für den der Sonnenstand berechnet werden soll.

10.2 Schnittstellenbeschreibung #20002



10.2.1 Ausgang Elevation

```
network output SNVT_angle_deg nvoElevation;
```

Die `nvoElevation` stellt anderen Knoten Information über die momentane Sonnenhöhe zur Verfügung.

Valid Range

<code>nvoElevation</code>	-359,98..+360° (0,02°)
---------------------------	------------------------

Default Value

Nach Reset des Gerätes hat `nvoElevation` den Werte:

<code>nvoElevation</code>	0x7FFF
---------------------------	--------

When Transmitted

Die Sendezeit für Elevation und Azimut werden gemeinsam über die `nviConfigTime` (Siehe 13.1.1) parametrisiert.

10.2.2 Ausgang Azimut

```
network output SNVT_angle_deg nvoAzimuth;
```

Die `nvoAzimuth` stellt anderen Knoten Information über die momentane Sonnenseitenwinkel zur Verfügung.

Valid Range

<code>nvoAzimuth</code>	-359,98..+360° (0,02°)
-------------------------	------------------------

Default Value

Nach Reset des Gerätes hat `nvoAzimuth` den Werte:

<code>nvoAzimuth</code>	0x7FFF
-------------------------	--------

10.3 Configuration Properties

10.3.1 Längengrad

```
network input config SNVT_angle_deg nciDegreeLongit;
```

Diese Konfigurationsvariable legt den Längengrad des Ortes fest für den die Sonnenstandsberechnung durchgeführt werden soll (Resource files: UCPTDegreeLongit).

Valid Range

<code>nciDegreeLongit</code>	-359,98..+360° (0,02°)
------------------------------	------------------------

Siehe auch [1].

Default Value

`nciDegreeLongit` = 580 (11,6° = München)

10.3.2 Breitengrad

```
network input config SNVT_angle_deg nciDegreeLatitu;
```

Diese Konfigurationsvariable legt den Breitengrad des Ortes fest für den die Sonnenstandsberechnung durchgeführt werden soll (Resource files: UCPTDegreeLatitu).

Valid Range

<code>nciDegreeLatitu</code>	-359,98..+360° (0,02°)
------------------------------	------------------------

Siehe auch [1].

Default Value

`nciDegreeLatitu` = 2405 (48,1° = München)

10.3.3 Differenz zur MEZ

network input UCPTDifferToMEZ nciDifferToMEZ;

Diese Konfigurationsvariable legt die Differenz zwischen Ortszeit und Mitteleuropäischer Zeit fest (Resource files: UCPTDifferToMEZ). Die Berechnung des Sonnenstandes beruht auf der MEZ (Mitteleuropäischen Zeit). Daher muß stets ein Bezug zur MEZ bereitgestellt werden.

Valid Range

nciDifferToMEZ	-32767...32767 (1 min)
----------------	------------------------

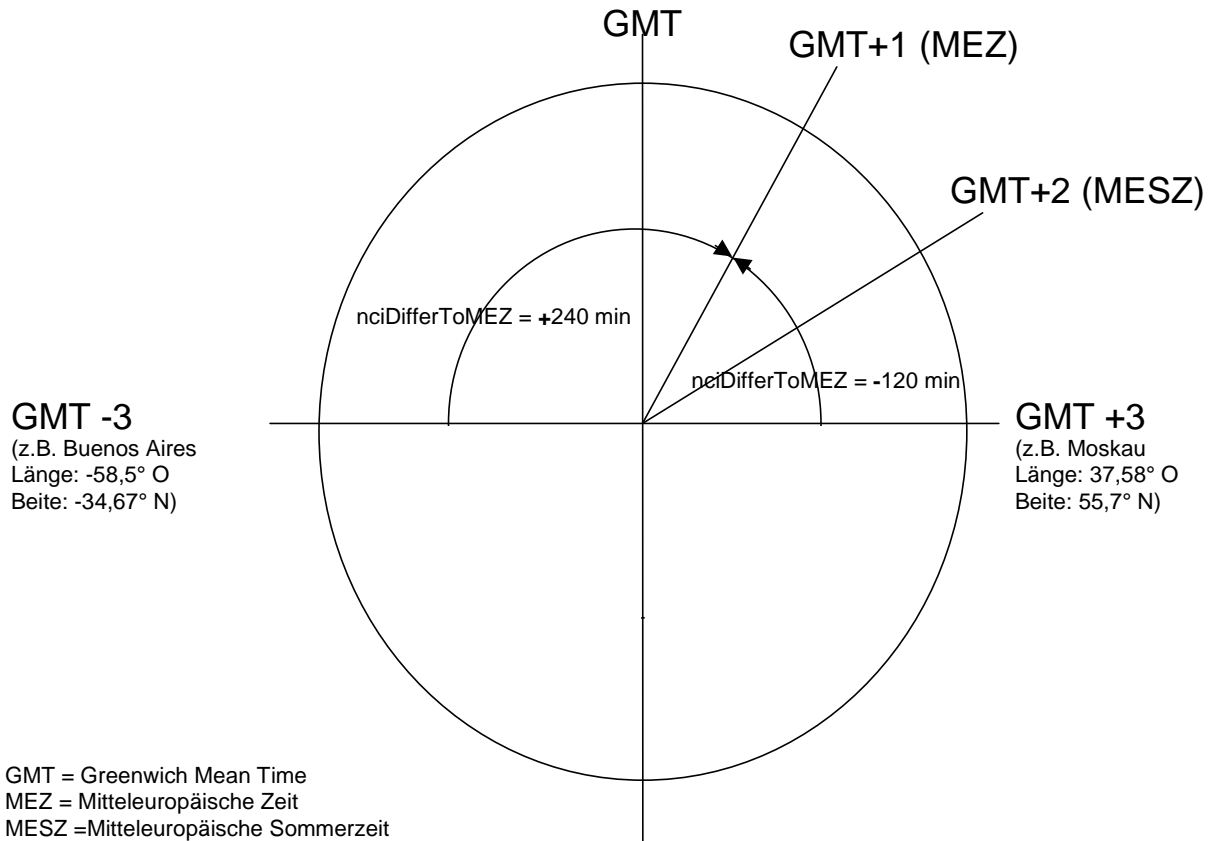


Abbildung 2, Zusammenhang Ortszeit / MEZ / GMT

Default Value

nciDifferToMEZ	0
----------------	---

Für die Parametrierung ist es also entscheidend ob der LONSE im Gültigkeitsbereich der MEZ betrieben wird und ob ein Empfang des Funkuhrsignals möglich ist.

Gültigkeitsbereich MEZ?	DCF77 Empfang?	Nötige Parametrierung!
NEIN	NEIN	Längen und Breitengrad parametrieren. Binden einer Externen Uhr mit Ortszeit auf <code>nviTimeSet</code> . Differenz zwischen Ortszeit und MEZ in <code>nciDifferToMEZ</code> eintragen.
NEIN	JA	Längen und Breitengrad parametrieren. Differenz zwischen Ortszeit und MEZ in <code>nciDifferToMEZ</code> eintragen.
JA	NEIN	Längen und Breitengrad parametrieren. Binden einer Externen Uhr mit Ortszeit auf <code>nviTimeSet</code> .
JA	JA	Längen und Breitengrad parametrieren.

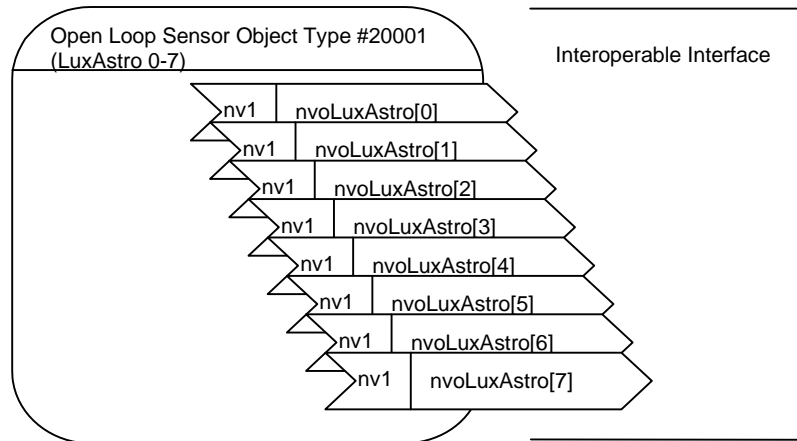
Um z.B. den Sonnenstand für Buenos Aires zu berechnen müßte neben den entsprechenden Winkeln für Längen- und Breitengrad `nciDifferToMEZ` mit dem Wert 240 (+4 Stunden) parametrisiert werden. Es wird hier vorausgesetzt das auf die Netzwerkvariable `nviTimeSet` eine Netzwerkvariable mit der Ortszeit Buenos Aires gebunden ist!

Ort	Längengrad	Breitengrad	Differenz zur MEZ
München	11,58°	48,13°	0 min
Stuttgart	9,20°	48,78	0 min
Hamburg	10,0°	53,55°	0 min
Berlin	13,43°	52,50°	0 min
Frankfurt (Main)	8,68°	50,10°	0 min
Leipzig	12,33°	51,33°	0 min
Wien (Österreich)	16,37°	48,22°	0 min
Zürich (Schweiz)	8,55°	47,38°	0 min
Paris (Frankreich)	2,33°	48,87°	0 min
Rom (Italien)	12,50°	41,88°	0 min
Amsterdam (Niederlande)	4,90°	52,35°	0 min
London (England)	-0,17°	51,50°	60 min
Washington (USA)	-77,02°	38,90°	360 min
Buenos Aires (Argentinien)	-58,50°	-34,67°	240 min
Moskau (GUS)	37,58°	55,75°	-120 min
Tokyo (Japan)	139,77°	35,70°	-480 min

Tabelle 2, Beispiele für mögliche Parametrierungen

11 Helligkeitssensor Astro #20001

11.1 Schnittstellenbeschreibung



11.1.1 Ausgang Helligkeit / Astro

network output UNVT_lux nvoLuxAstro[X];

Diese Variable stellt auf dem Netzwerk einen kombinierten Helligkeits- und Sonnenstandswert zur Verfügung (Resource files: UNVT_lux)

Valid Range

Element in	Bedeutung
nvoLuxAstro[X]	
.unsigned long	0..65535 (2 lux) Helligkeitswert
.SNVT_angle_deg	-359,98..369° (0,02°) Elevation
.SNVT_angle_deg	-359,98..369° (0,02°) Azimut

Default Value

Nach Reset des Gerätes hat nvoLuxAstro[X] den Wert:

Element in	
nvoLuxAstro[X]	
.unsigned long	0xFFFF
.SNVT_angle_deg	0x7FFF
.SNVT_angle_deg	0x7FFF

Beispiel

Wenn die Netzwerkvariable unformatiert angezeigt wird:
Helligkeit 15 kLux, 25 Grad Elevation, 100 Grad Azimut setze nvoLuxAstro
auf 1D 4C 04 E2 13 88.

When Transmitted

Die MaxSendTime und MinSendTime für die Netzwerkvariable `nvoLuxAstro[X]` wird über die `nviConfigTime` (Siehe 13.1.1) parametrieren. Alle sechs Helligkeitsvariablen werden über einen Parametersatz parametrieren!

11.2 Configuration Properties

Keine.

12 Eisüberwachung

Dieses Objekt Realisiert eine Eisüberwachung unter Berücksichtigung von Außentemperatur und Niederschlag.

12.1 Allgemeines zur Eisüberwachung

Die Netzwerkvariable `nvoIce` wird aus den Meßwerten Temperatur Außen UND Niederschlag gebildet. Es ist daher nötig das die entsprechenden Sensoren angeschlossen sind!

12.2 Funktion

Zustand Eisalarm herrscht, falls `nciIceTemp` unterschritten ist und Niederschlag erkannt wird. Zustand Eisalarm wird aufgehoben, falls die Außentemperatur den Grenzwert `nciIceTemp` überschreitet. Ein Anstehender Eisalarm wird auch nach einem „Reset“ des Gerätes (z.B. Spannungsausfall) aufrecht erhalten bis die Außentemperatur `nciIceTemp` überschreitet!

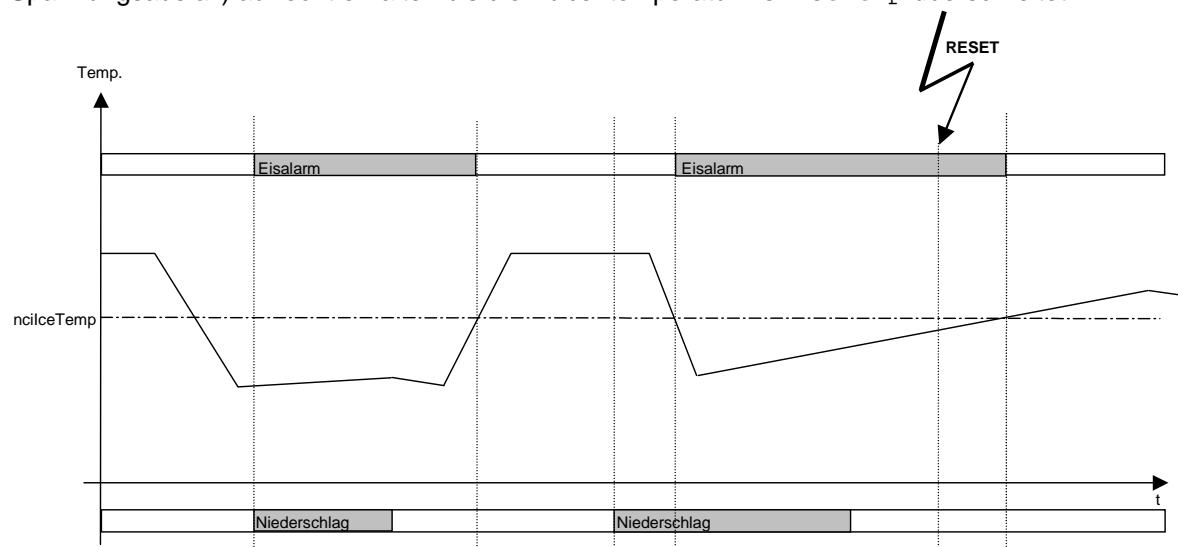
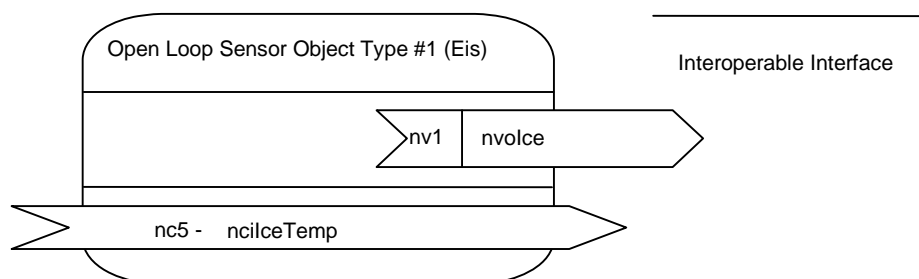


Abbildung 3, Eisalarm: Zusammenhang Temperatur und Niederschlag

12.3 Schnittstellenbeschreibung



12.3.1 Ausgang Eisüberwachung

```
network output SNVT_switch nvoIce;
```

Die `nvoIce` stellt anderen Knoten Information über einen möglichen Eisalarm zur Verfügung.

Valid Range

Element in	Kein Eisalarm	Eisalarm
nvoIce		
.value	0x00	0xC8
.state	0x00	0x01

Default Value

Nach Reset des Gerätes hat nvoIce die Werte:

Element in	
nvoIce	
.value	0x00 (Bei Eisalarm vor Reset 0xC8)
.state	0x00 (Bei Eisalarm vor Reset 0x01)

When Transmitted

Die MaxSendTime und MinSendTime für die Netzwerkvariable nvoIce wird über die nviConfigTime (Siehe 13.1.1) parametrisiert.

12.4 Configuration Properties

12.4.1 Schaltschwelle Eisüberwachung

```
network input config SNVT_temp_p nciIceTemp;
```

Diese Konfigurationsvariable legt die Schaltschwelle fest welche die Außentemperatur über- bzw. unterschreiten muß um das Auslösen eines Eisalarms bzw. das Zurücknehmen eines Eisalarms zu bewirken.

Valid Range

nciIceTemp	-273,17..327,66°C (0,01°C)
------------	----------------------------

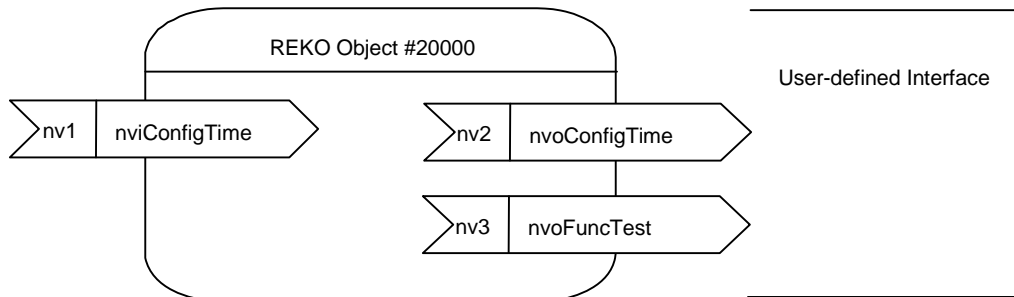
Siehe auch [1].

Default Value

nciIceTemp = 300 (3°C)

13 REKO Object #20000

13.1 Schnittstellenbeschreibung



13.1.1 Eingang Sendezeiten Netzwerkvariablenverwaltung

```
network input UNVT_time_cnf nviConfigTime;
```

Die Netzwerkvariable UNVT_time_cnf ist eine „User-defined“ Netzwerkvariable.

```
typedef struct tagNvAdminTime
{
    unsigned int uiObjectService;
    unsigned int uiObjectNumber;
    SNVT_time_sec MaxSendTime;
    SNVT_time_sec MinSendTime;
} UNVT_time_cnf;
```

Diese Variable dient zur Parametrierung der Sendezeiten der verschiedenen Objekte des Netzwerkknoten (Resource files: UNVT_time_cnf).

Element in:	Bereich:	Bedeutung:
nviConfigTime		
.uiObjectService	0 SAVE 1 CLEAR 2 REPORT -1 NUL	Speichert Zeiten „nicht flüchtig“ Nicht implementiert Meldet Zeiten über nvoConfigTime zurück Ungültiger Dienst
.uiObjectNumber	0..127	Siehe unten
.MaxSendTime	0..65535	Maximale Zeit zwischen zwei NV-Sendungen (0,1 s)
.MinSendTime	0..65535	Minimale Zeit zwischen zwei NV-Sendungen (0,1 s)

Über uiObjectService wird ein Dienst ausgewählt. Mit uiObjectNumber wird bestimmt für welche Objekte der Dienst ausgeführt werden soll. Die Sendezeiten MaxSendTime und MinSendTime werden, wenn der Dienst SAVE ausgewählt ist, „nicht flüchtig“ gespeichert und bestimmen dann die Sendezeiten der Netzwerkvariablen der Objekte. Die Parametrierung der Sendezeiten ermöglicht Einstellungen zwischen 1 Sekunde (1) und 109 Minuten (65565). Hier ist darauf zu achten das lediglich ein 1 Sekunden-Raster ausgewertet wird; ein Wert von 105 entspricht also 10 Sekunden.

Wird der Dienst REPORT gewählt, so wird lediglich die aktuelle MaxSendTime und MinSendTime für die ausgewählten Objekte über nvoConfigTime zurückgemeldet. Ist für ein Objekt nur eine MaxSendTime implementiert, so wird der MinSendTime Wert verworfen.

Objekt Nr.	Funktion	MaxSend Time	Default Value	MinSend Time	Default Value	uiObject Number
0	Node Object	NEIN	--	NEIN	--	0
1	Temperatur, nvoTemp	JA	600	JA	200	1
2	Relative Luftfeuchtigkeit nvoRelHumid	JA	600	JA	200	2
3	Windrichtung nvoWindDirection	JA	600	JA	200	3
4-11	Helligkeit nvoLux[0-7]	JA	600	JA	200	4
12-14	Windgeschwindigkeit nvoWindSpeed[0-2]	JA	200	JA	30	12
15	Niederschlag nvoRain	JA	600	JA	200	15
16	Funkuhr nvoTimeDate	JA	600	NEIN	--	16
	Wochentag und MEZ/MESZ nvoDayDate nvoMEZMESZ	JA	600	NEIN	--	127
17	Sonnenstandsberechnung nvoElevation nvoAzimuth	JA	600	NEIN	--	17
18-25	Helligkeit/Sonnenstand nvoLuxAstro[0-7]	JA	600	NEIN	--	18
26	Eisüberwachung nvoIce	JA	600	JA	200	26
27	REKO Object	NEIN	--	NEIN	--	27

Tabelle 3, Zuordnung Objekte zu Sendezeiten und Default Values

Beispiele Schreiben:

Element in nviConfigTime	
.uiObjectService	0
.uiObjectNumber	1
.MaxSendTime	500
.MinSendTime	200

Ausgewählter Dienst ist Schreiben. Geschrieben werden soll die Sendezeit des Objekts 1 (Temperatur Sensor). MaxSendTime sind 50 Sekunden, MinSendTime sind 20 Sekunden.

Wenn die nviConfigTime nicht formatiert dargestellt wird:

nviConfigTime 00 01 01 F4 00 C8

Element in nviConfigTime	
.uiObjectService	0x00
.uiObjectNumber	0x01
.MaxSendTime	0x01 0xF4
.MinSendTime	0x00 0xC8

13.1.2 Ausgang Sendezeiten Netzwerkvariablenverwaltung

network output UNVT_time_cnf nvoConfigTime;

Diese Variable dient zur Rückmeldung der Parametrierung der Sendezeiten der verschiedenen Objekte des Netzwerkknoten (Resource files: UNVT_time_cnf).

Element in	Bedeutung
nvoConfigTime	
.uiObjectService	0 SAVE (Speichert Zeiten „nicht flüchtig“) 1 CLEAR (Nicht implementiert) 2 REPORT (Meldet Zeiten über nvoConfigTime zurück) -1 NUL (Ungültiger Dienst)
.uiObjectNumber	0..127
.MaxSendTime	0..65535
.MinSendTime	0..65535

Über uiObjectService wird der zuletzt angeforderte Dienst angezeigt. Mit uiObjectNumber wird angezeigt für welche Objekte der Dienst ausgeführt wurde. MaxSendTime und MinSendTime werden angezeigt, für Objekte die nur über eine MaxSendTime verfügen wird als MinSendTime hier der Maximalwert ausgegeben.

Beispiele Lesen:

Element in	
nviConfigTime	
.uiObjectService	2
.uiObjectNumber	3
.MaxSendTime	12345
.MinSendTime	65432

Lesen der Sendezeiten des Objekts 3 (Windrichtung Sensor). MaxSendTime und MinSendTime die beim Lesen hier eingetragen sind haben KEINE Bedeutung. Die angeforderten Werte stehen danach in nvoConfigTime zur Verfügung.

Wenn die nvoConfigTime unformatiert dargestellt wird:

nvoConfigTime 02 03 30 39 FF 98

Element in	
nviConfigTime	
.uiObjectService	0x02
.uiObjectNumber	0x03
.MaxSendTime	0x30 0x39
.MinSendTime	0xFF 0x98

Die Abbildung 4 soll das Sendeverhalten der Netzwerkvariablen verdeutlichen. Prinzipiell kann zu drei verschiedenen Zeitpunkten gesendet werden.

- Es wird spätestens nach Ablauf von MaxSendTime gesendet
- Es wird sofort nach Ablauf von MinSendTime gesendet wenn innerhalb von MinSendTime der SendOnDelta Wert überschritten wurde
- Es wird nach Ablauf von MinSendTime innerhalb MaxSendTime gesendet wenn der SendOnDelta Wert überschritten wird

Als SendOnDelta Wert wird die Differenz zwischen dem zuletzt gesendeten Wert und dem vom Knoten aktuell ermittelten Wert bezeichnet.

Die maximale Sendehäufigkeit wird also durch MinSendTime bestimmt, die minimale Sendehäufigkeit wird durch MaxSendTime bestimmt. Ist keine MinSendTime vorgesehen, wie zum Beispiel bei der Uhrzeit (nvoTimeDate), wird immer nach Ablauf von MaxSendTime gesendet.

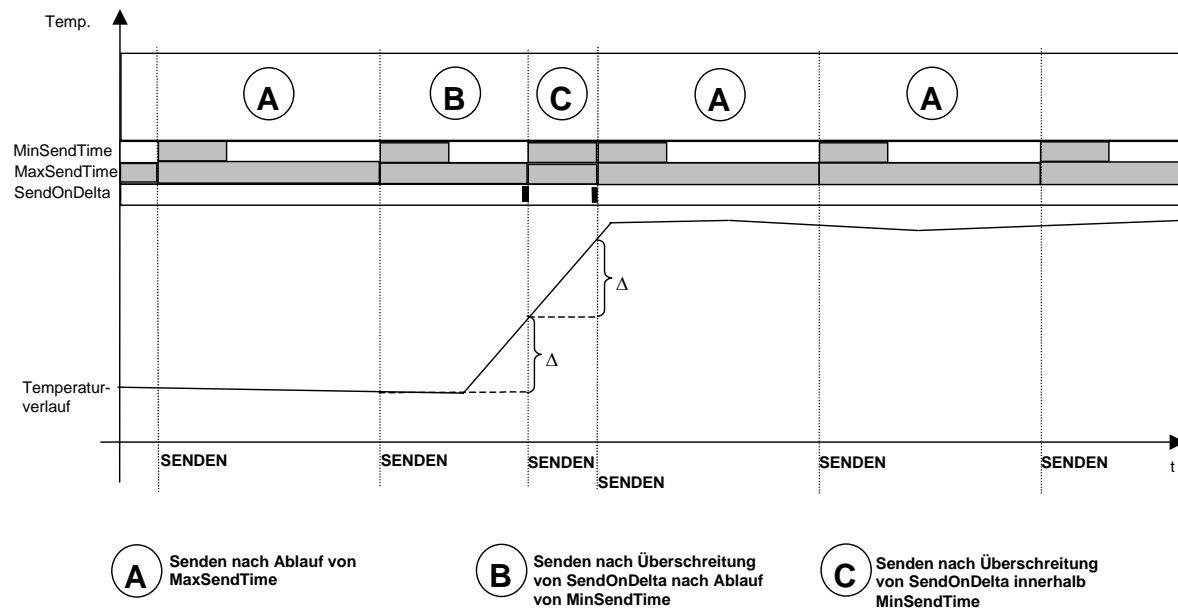


Abbildung 4, Graphische Darstellung des Sendeverhaltens

13.1.3 Ausgang Funktionstest

network output UNVT_state nvoFuncTest;

Diese Variable bietet die Möglichkeit die Eingänge des LONSE zu überprüfen (Resource files: UNVT_state)..

Elemente in nvoFuncTest .bit	Sensor	Funktion
0	Foto1	0 wenn Eingang 0V, 1 wenn Eingang 12V
1	Foto2	0 wenn Eingang 0V, 1 wenn Eingang 12V
2	Foto3	0 wenn Eingang 0V, 1 wenn Eingang 12V
3	Foto4	0 wenn Eingang 0V, 1 wenn Eingang 12V
4	Foto5	0 wenn Eingang 0V, 1 wenn Eingang 12V
5	Foto6	0 wenn Eingang 0V, 1 wenn Eingang 12V
6	Foto7	0 wenn Eingang 0V, 1 wenn Eingang 12V
7	Foto8	0 wenn Eingang 0V, 1 wenn Eingang 12V
8	Temp.	0 wenn Eingang 0V, 1 wenn Eingang 12V
9	Feucht.	0 wenn Eingang 0V, 1 wenn Eingang 12V
10	Windrichtg.	Wechselt jeweils seinen Zustand wenn von „West“ nach „Ost“ gewechselt wird
11	Niederschl.	0 wenn Eingang 0V, 1 wenn Eingang 12V
12	Funkuhr	Wechselt jeweils bei Anlegen von 12V bzw. 0V seinen Zustand bzw. wenn DCF77 Empfänger angeschlossen ist wechselt Ausgangszustand sekundlich automatisch
13	Wind1	Wechselt jeweils seinen Zustand wenn Flügelrad gedreht wird
14	Wind2	Wechselt jeweils seinen Zustand wenn Flügelrad gedreht wird
15	Wind3	Wechselt jeweils seinen Zustand wenn Flügelrad gedreht wird

Um die Eingänge zu testen kann z.B. Betriebsspannung (12V_) an die entsprechenden Klemme angelegt und wieder entfernt werden. Das entsprechende Bit der Netzwerkvariable nvoFuncTest muß dann seinen Zustand ändern. Dies soll die Inbetriebnahme und Fehlersuche erleichtern.

14 Quellenangaben

- [1] SNVT Master List and Programmers Guide, Echelon Corporation, May 1997
- [2] LONMARK Application Layer Interoperability Guidelines Version 3.1, LONMARK Interoperability Association, 1992-1998
- [3] NEURON-C Reference Guide
- [4] MOTOROLA LonWorks Technology Device Data
- [5] Benutzer und Installationsanleitung Sensoreinheit LONSE (B&I)