

NTP-LIB

SNTP Zeitsynchronisation unter RTOS-UH

Dok-Rev. 1.7 vom 08.11.2018

Software-Rev. 1.3 vom 04.05.2007

Inhaltsverzeichnis

1	Urheberrecht und Haftung	4
2	NTP unter RTOS-UH.....	5
2.1	Die Voraussetzungen für NTP	5
2.1.1	Environmentvariable NTP_SERVER	5
2.1.2	Environmentvariable UTC_OFFSET	5
2.1.3	Environmentvariable DAY_LIGHT	5
2.1.4	Environmentvariable NTP_INTERVAL	6
2.1.5	Environmentvariable NTP_MAX_DIFF	6
2.1.6	Environmentvariable NTP_MAX_REGEL	6
3	Der NTP-Client.....	7
4	Die NTP-LIB Libraries	8
4.1	Funktionen der NTP-Library im Überblick	8
4.1.1	NTP_STOP_PROC	9
4.1.2	NTP_STATUS_PROC	9
4.1.3	NTP_SET_STANDORT	9
4.1.4	NTP_UTC_OFFSET	10
4.1.5	NTP_DAY_LIGHT	10
4.1.6	NTP_GET_SERVER	10
4.1.7	NTP_START_PROC	11
4.1.8	NTP_SEARCH_PROC	12
4.1.9	NTP_PROC_STATISTIC	13
5	Der NTP-Server	14
6	Automatische Sommer/Winterzeit Umstellung	15
6.1	CHECK_AND_SET_SUMMERTIME	15
6.2	GET_AKT_WOCHENTAG	15
6.3	Beispiel Zeitumstellung	16

Revisionsliste:

Rev.	Datum	Na.	Änderung
1.0	20.07.2006	Kr	Erstellung
1.1	30.01.2007	Kr	NTP_SHOW_STATISTIC ausführlicher beschrieben
1.2	01.02.2007	Kr	In der ResyncStatistic neuen Code eingeführt
1.3	06.02.2007	Ko	Ergänzungen
1.4	29.03.2007	Kr	Environment Var DAY_LIGHT für Sommer/Winterzeit
1.5	10.04.2007	Kr	NTP_SRV unter RTOS-UH
1.6	04.05.2007	Kr	Automatische Sommer/Winterzeitumstellung
1.7	14.11.2007	Ko	Korrekturen
1.8	08.11.2018	Ha	NTP_PROC_STATISTIK: Liste der Grund-Werte erweitert

1 Urheberrecht und Haftung

Alle Rechte an diesen Unterlagen liegen bei der IEP GmbH, Langenhagen.

Die Vervielfältigung, auch auszugsweise, ist nur mit unserer ausdrücklichen schriftlichen Genehmigung zulässig.

In Verbindung mit dem Kauf von Software erwirbt der Käufer einfaches, nicht übertragbares Nutzungsrecht. Dieses Recht zur Nutzung bezieht sich ausschließlich darauf, daß dieses Produkt auf oder in Zusammenhang mit jeweils **einem** Computer zu benutzen ist. Das Erstellen einer Kopie ist ausschließlich zu Archivierungszwecken unter Aufsicht des Käufers oder seines Beauftragten zulässig. Der Käufer haftet für Schäden, die sich aus der Verletzung seiner Sorgfaltspflicht ergeben, z.B. bei unautorisiertem Kopieren, unberechtigter Weitergabe der Software usw.. Der Käufer gibt mit dem Erwerb der Software seine Zustimmung zu den genannten Bedingungen. Bei unlizenziertem Kopieren muß vorbehaltlich einer endgültigen juristischen Klärung von Diebstahl ausgegangen werden. Dies gilt ebenso für Dokumentation und Software, die durch Modifikation aus Unterlagen und Programmen von IEP hervorgegangen ist, gleichgültig, ob die Änderungen als geringfügig oder erheblich anzusehen sind.

Eine Haftung seitens IEP für Schäden, die auf den Gebrauch von Software, Hardware oder Benutzung dieses Manuskriptes zurückzuführen sind, wird ausdrücklich ausgeschlossen, auch für den Fall fehlerhafter Software oder irrtümlicher Angaben.

Das Einverständnis des Käufers oder Nutzers für den Haftungsausschluß gilt mit dem Kauf und der Nutzung der Software und dieser Unterlagen als erteilt.

2 NTP unter RTOS-UH

Die NTP Library und der NTP-Client dienen zur Zeitsynchronisation der RTOS-UH internen Uhr mit einem NTP-Server.

2.1 Die Voraussetzungen für NTP

Für die volle Funktionalität müssen im System folgende Voraussetzungen geschaffen sein:

1. Es muss eine globale Environmentvariable `NTP_SERVER` geben, die die IP-Adresse eines erreichbaren NTP-Servers angibt.
2. Es sollte eine globale Environmentvariable `UTC_OFFSET` geben, die den Standortoffset des RTOS-UH Rechners gegenüber UTC angibt. Der Offset kann in Stunden (± 24) bzw. Minuten (>60) angegeben werden.
3. Es sollte eine globale Environmentvariable `DAY_LIGHT` geben, die die Sommer/Winterzeit angibt. Der Zeitversatz zwischen Sommer und Winterzeit kann in Stunden (± 24) bzw. Minuten (>60) angegeben werden.

Wird der NTP-Client genutzt, ist keine eigene Programmierung nötig, es müssen nur die obigen Environmentvariablen gesetzt werden. Für die Einbindung in eigene Programme steht eine PEARL-Library zur Verfügung.

2.1.1 Environmentvariable `NTP_SERVER`

Die Environmentvariable `NTP_SERVER` wird folgendermaßen gesetzt:

```
ENVSET -G NTP_SERVER=192.168.10.10
```

Sie gibt an, unter welche IP-Adresse der im Netzwerk verfügbare NTP-Server angesprochen werden kann.

2.1.2 Environmentvariable `UTC_OFFSET`

Die Environmentvariable `UTC_OFFSET` wird folgendermaßen gesetzt:

```
ENVSET -G UTC_OFFSET=1
```

Sie gibt den Zeitoffset der RTOS-Clock gegenüber UTC an, hier +1 Stunde.

2.1.3 Environmentvariable `DAY_LIGHT`

Die Environmentvariable `DAY_LIGHT` wird folgendermaßen gesetzt:

```
ENVSET -G DAY_LIGHT=1
```

Sie gibt Sommer(1)/Winter(0)zeit an, hier +1 Stunde (Sommerzeit).

2.1.4 Environmentvariable NTP_INTERVAL

Die Environmentvariable NTP_INTERVAL wird folgendermaßen gesetzt:

```
ENVSET -G NTP_INTERVAL=16
```

Sie gibt an, in welchem Zeitintervall der NTP-Server abgefragt werden soll. Die Angabe erfolgt in Sekunden.

2.1.5 Environmentvariable NTP_MAX_DIFF

Die Environmentvariable NTP_MAX_DIFF wird folgendermaßen gesetzt:

```
ENVSET -G NTP_MAX_DIFF=100
```

Hiermit wird festgelegt, ab welcher Zeitabweichung (in ms) zwischen der RTOS-Clock und dem NTP-Server die RTOS-Clock mittels eines CLOCKSET-Kommandos **hart** synchronisiert wird. Normalerweise wird die RTOS-Clock **weich** nachgeregelt. Dabei wird der Nonius der RTOS-Clock verstellt, so dass eine **harte** Synchronisation nicht notwendig ist. Wichtig ist die Angabe von NTP_MAX_DIFF für die Synchronisierung beim Start des NTP-Clients sowie bei Zeitwechsel des NTP-Servers, z.B. bei Schaltsekunden.

Bei allen RTOS-UH-Systemen, deren Clock ohne Nonius läuft, wird bei Erreichung dieser Grenze die RTOS-Clock um die mit NTP_MAX_REGEL angegebene Zeit verstellt.

2.1.6 Environmentvariable NTP_MAX_REGEL

Die Environmentvariable NTP_MAX_REGEL wird folgendermaßen gesetzt:

```
ENVSET -G NTP_MAX_REGEL=10
```

Sie gibt an, ab welchem Clockoffset (im ms) die RTOS-Clock verstellt wird, um die Zeitabweichung zum NTP-Server auszugleichen.

Bei RTOS-Systemen mit Nonius wird die Zeitangabe auf 0 gesetzt, da sie nicht berücksichtigt werden muss.

3 Der NTP-Client

Es steht ein NTP-Client (SNTPCLI . SR) zur Verfügung, der mit den beschriebenen Environmentvariablen konfiguriert wird. Folgende Tasks stehen zur Verfügung:

Task	Funktionalität
NTP_TIME_SYNC	Führt die Zeitsynchronisation durch.
NTP_STAT	Ausgabe des Statuses des NTP-Servers
NTP_SOMMERZEIT	Zeit = UTC_OFFSET + 1 HRS
NTP_WINTERZEIT	Zeit = UTC_OFFSET
NTP_START	Startet Zeitsynchronisation mit hartem CLOCKSET – Eingriff
NTP_START_NO_REGEL	Starten der Zeitsynchronisation ohne harten CLOCKSET Eingriff (siehe NTP_START_PROC)
NTP_SEARCH	Anzeige verfügbarer NTP-Server
NTP_SHOW_STATISTIC	Anzeige der Synchronisationsstatistik
NTP_STOP	Stop der Zeitsynchronisierung

Defaulteinstellungen sind:

- 500 ms maximale Zeitabweichung
- 10 s Synchronisationsintervall mit dem NTP-Server
- max_clockset = 0 bei NTP_START_NO_REGEL, d.h. keine Verstellung mit CLOCKSET –T

Mit diesen Einstellungen läuft RTOS-UH im Regelfall zeitsynchronisiert, so dass die Programmierung eines eigenen NTP-Clients nicht notwendig ist.

4 Die NTP-LIB Libraries

Für die Sprache PEARL steht die Library `SNTPLIB.SR` zur Verfügung, diese kann auch permanent im Eprom oder Flash abgelegt werden.

4.1 Funktionen der NTP-Library im Überblick

Für die Programmierung eines eigenen NTP-Client stehen in der Library folgende Funktionen zur Verfügung:

Funktion	Bedeutung
<code>NTP_STOP_PROC</code>	Stoppen der Zeitsynchronisation
<code>NTP_STATUS_PROC</code>	Auslesen des Status des NTP-Servers
<code>NTP_SET_STANDORT</code>	Setzen des UTC_Offsets
<code>NTP_UTC_OFFSET</code>	Einlesen der ENV-Variable <code>UTC_OFFSET</code>
<code>NTP_GET_SERVER</code>	Einlesen der ENV-Variable <code>NTP_SERVER</code>
<code>NTP_START_PROC</code>	Starten der Zeitsynchronisation
<code>NTP_SEARCH_PROC</code>	Anzeige verfügbarer NTP-Server
<code>NTP_PROC_STATISTIC</code>	Anzeige der Synchronisationsstatistik
<code>NTP_DAY_LIGHT</code>	Einlesen der ENV-Variable <code>DAY_LIGHT</code>

4.1.1 NTP_STOP_PROC

Die Funktion NTP_STOP_PROC stoppt die Task NTP_TIME_SYNC.

```
SPC NTP_STOP_PROC GLOBAL ;
...
CALL NTP_STOP_PROC ;
...
```

4.1.2 NTP_STATUS_PROC

Mit dem Aufruf der Funktion NTP_STATUS_PROC wird der aktuelle Status des NTP-Servers aus der Library zurückgegeben. Der Status ist eine BIT(16) Variable mit folgenden Bedeutung der Einzelbits:

```
ret.BIT(1)    NTP started
ret.BIT(2)    NTP-Server Online
ret.BIT(5..12) Stratum des NTP-Servers
               0x00 = unspecified
               0x01 = primary reference (e.g. radio clock)
               0x02-0x0F = secondary reference (via NTP or SNTP)
               0x10-0xFF = reserved
ret.BIT(15..16) State of NTP-Server (LI)
               0 = synchron
               3 = not synchron
```

```
SPC NTP_STATUS_PROC ENTRY RETURNS( BIT(16) ) GLOBAL ;
...
DCL stat BIT(16) ;
...
stat = NTP_STATUS_PROC ;
```

4.1.3 NTP_SET_STANDORT

Die Funktion NTP_SET_STANDORT setzt den aktuellen UTC_offset

```
SPC NTP_SET_STANDORT ENTRY( offs FIXED(31) ) GLOBAL ;
...
DCL offset FIXED(31) INIT(3(31)) ; /* Moskau Zeit */
...
CALL NTP_SET_STANDORT( offset ) ;
```

4.1.4 NTP.UTC_OFFSET

Die Funktion NTP.UTC_OFFSET liest die Environmentvariable UTC_OFFSET aus und liefert deren Werte zurück. Falls die ENV-Variable nicht gefunden wird, ist der Wert = 0.

```
SPC NTP.UTC_OFFSET ENTRY RETURNS( FIXED(31) ) GLOBAL ;
...
DCL glb_offset  FIXED(31) ;
glb_offset = NTP.UTC_OFFSET;
```

4.1.5 NTP.DAY_LIGHT

Die Funktion NTP.DAY_LIGHT liest die Environmentvariable DAY_LIGHT aus und liefert deren Werte zurück. Falls die ENV-Variable nicht gefunden wird, ist der Wert = 0.

```
SPC NTP.DAY_LIGHT ENTRY RETURNS( FIXED(31) ) GLOBAL ;
...
DCL glb_day_light  FIXED(31) ;
glb_day_light = NTP.DAY_LIGHT;
```

4.1.6 NTP.GET_SERVER

Die Funktion NTP.GET_SERVER liest die Environmentvariable NTP_SERVER aus und liefert deren Wert und die zugehörige Broadcast-Adresse zurück. Falls die ENV-Variable nicht gefunden wird, werden die übergebenen Parameter nicht verändert.

```
SPC NTP.GET_SERVER ENTRY
( broadcast BIT(32) IDENT ,
  srv BIT(32) IDENT ) GLOBAL ;
...
/* Variablen auf default setzen */
DCL the_broadcast BIT(32) INIT('FFFFFF00'B4) ;
DCL the_srv      BIT(32) INIT('C0A80A0A'B4) ;
...
CALL NTP.GET_SERVER( the_broadcast, the_srv ) ;
```

4.1.7 NTP_START_PROC

Die Funktion NTP_START_PROC startet den Zeitsynchronisationsvorgang. Sie parametrisiert die Library und startet in der Library die Task NTP_TIME_SYNC, die die eigentliche Zeitsynchronisation durchführt. Die Zeitsynchronisierung wird mit einer Noniusverstellung der RTOS-Clock im Feinbereich durchgeführt und nur bei grösseren Zeitabweichungen erfolgt eine absolute Zeitverstellung.

Die Bedeutung der einzelnen Parameter ist wie folgt:

max_abw: Bei einer grösseren Abweichung (in ms) zwischen der RTOS-Clock und dem NTP-Server wird die RTOS-Clock hart mittels eines

CLOCKSET -T xx:xx:xx.xxx

auf die NTP-Zeit gestellt.

utc_abw: Dieses ist die Abweichung der RTOS-Clock gegenüber UTC in Stunden, falls der Wert zwischen -24 und 24 liegt. Falls der Wert vom Betrag ≥ 60 ist, ist der Offset in Minuten angegeben.

max_clockset: Falls dieser Parameter ungleich von 0(31) angegeben wird, dann wird auch bei kleinerer Abweichung als **max_abw**, aber grösser als **max_clockset** die RTOS-Clock mit einem „CLOCKSET -T“ nachgestellt, aber maximal um die mit max_clockset angegebenen Grösse.

syn_time: Die Zeitspanne für das Abfrageintervall beim NTP-Server.

the_srv: Die IP-Adresse des NTP-Servers.

the_broadcast: Die Broadcast-Adresse für das Netz des Servers, wird nur für die Funktion NTP_SEARCH_PROC gesetzt und dort benötigt.

```
SPC NTP_START_PROC PROC ( the_broadcast          BIT(32) ,
    the_srv          BIT(32) ,
    max_abw          FIXED(31) ,
    utc_abw          FIXED(31) ,
    max_clockset     FIXED(31) ,
    syn_time         DURATION ) GLOBAL ;
...
/* Variablen auf default setzen */
DCL the_broadcast BIT(32) INIT('FFFFFF00'B4) ;
DCL the_srv       BIT(32) INIT('C0A80A0A'B4) ;
DCL utc_abw       FIXED(31) INIT(1(31)) ; /* MEZ */
DCL max_abw       FIXED(31) INIT(500(31)) ; /* 500ms */
DCL max_clockset  FIXED(31) INIT(0(31)) ;
DCL syn_time      DURATION INIT( 10 SEC ) ;
...
CALL NTP_START_PROC( the_broadcast, the_srv,
                    max_abw, utc_abw, max_clockset, syn_time
                    ) ;
```

4.1.8 NTP_SEARCH_PROC

Die Funktion NTP_SEARCH_PROC liefert die Daten aller unter dieser Broadcast-Adresse erreichbaren NTP-Server mit ihren relevanten Daten zurück.

```
SPC NTP_SEARCH_PROC PROC ( broadcast BIT(32),  
                           outp DATION INOUT ALPHIC IDENT ) GLOBAL;  
...  
CALL NTP_SEARCH_PROC('C0A80AFF'B4, TY ) ;  
...
```

Es ergibt sich z.B. folgende Ausgabe:

```
IP:C0A80A0A V4 LI:0 ST: 2 Pr:0.00000381 Dly: 0:00:00.000 Ofs: 0:00:00.000  
IP:C0A80A8D V4 LI:0 ST:15 Pr:0.00000381 Dly: 0:00:00.000 Ofs:-0:00:00.109  
IP:C0A80A62 V4 LI:0 ST: 2 Pr:0.00000381 Dly: 0:00:00.000 Ofs: 0:00:00.004  
IP:C0A80A0C V4 LI:0 ST: 1 Pr:0.00000381 Dly: 0:00:00.008 Ofs:-0:00:00.007  
IP:C0A80A6F V3 LI:3 ST: 0 Pr:0.00000381 Dly: 0:00:00.025 Ofs: 0:00:00.003  
IP:C0A80AA1 V3 LI:3 ST: 0 Pr:0.00000381 Dly: 0:00:00.037 Ofs:-0:00:00.141
```

Es werden alle NTP-Server mit ihrem Status sowie der Laufzeit und dem Time-Offsets zum lokalen Rechner ausgegeben. Die Ausgabezeile ist folgendermaßen aufgebaut:

IP : IP-Adresse
Vx: SNTP-Level
LI: 0 = synchronisiert , 3 nicht synchron
ST: Stratum des NTP-Servers, Bereich 1...15, je kleiner desto besser, aber 0=Fehler
Pr: Precision (Genauigkeit in Sekunden)
Dly: Laufzeit der NTP-Anfrage zu diesem Server
Ofs: Offset des NTP-Servers zu diesem Rechner

4.1.9 NTP_PROC_STATISTIC

Ausgabe von Statistikdaten des NTP-Servers und Synchronisierungszeiten zu diesem Rechner.

```
SPC NTP_PROC_STATISTIC PROC (
    outp DATION INOUT ALPHIC IDENT ) GLOBAL;
...
CALL NTP_PROC_STATISTIC( TY ) ;
...
```

Es ergibt sich z.B. folgende Ausgabe:

```
Online:1 Srv: C0A80A0A MaxDiff: 200 MaxReg: 0 ReSync:11:41:05.401
Counts: 184 Actual: -5 Min: -50 Max: 9 Avg: -8.71
        Statistik in [msec] Last AVG over [ 60 sec] -5.67
      <- 50 <- 10          >+ 10 >+ 50
        1      45      138      0      0
--Resync Statistic---<START>
30-01-2007 9:21:21.972 2 -0 HRS 00 MIN 00.791 SEC 1
30-01-2007 9:21:22.072 1 -0 HRS 00 MIN 00.001 SEC 0
--Resync Statistic---<END>
```

alle Angaben sind in ms.

In der ersten Zeile wird der Status ausgegeben, Sie sehen den Online-Status, die IP-Adresse des NTP-Servers, die Werte für MaxDiff und MaxReg sowie den Zeitpunkt der letzten Synchronisation. In der zweiten Zeile steht die Anzahl der Serveraufrufe sowie die Abweichung in ms für den aktuellen Abruf, den tiefsten, den höchsten und den Mittelwert der Abweichung. In der dritten Zeile findet sich der Mittelwert der Abweichung über die letzten 60 s. In den nächsten beiden Zeilen finden Sie die Anzahl der Aufrufe mit Abweichung innerhalb ± 10 ms, größer ± 10 ms oder größer ± 50 ms.

Die Bedeutung der Angaben in der Resync Statistik sind wie folgt:

Datum	Uhrzeit	Grund	Server-Offset	Server-Delay
-------	---------	-------	---------------	--------------

Grund kann folgende Werte enthalten:

- 2 : Server aus der Synchronisation gefallen LI=3 oder Stratum=0
- 1 : Server ist nicht erreichbar
- 1 : Synchronisation innerhalb der gewünschten Toleranz
- 2 : Nachsynchronisation mittels Uhrverstellung, da ausserhalb der Toleranz
- 3 : Synchronisation innerhalb des 3-fachen der Umgebungsvariable NTP_MAX_DIFF
- 4 : Synchronisation ausserhalb des 3-fachen der Umgebungsvariable NTP_MAX_DIFF

5 Der NTP-Server

Es steht ein NTP-Server (SNTPSRV.SR) unter RTOS-UH zur Verfügung, der mit den beschriebenen Environmentvariablen konfiguriert wird. Es müssen die ENV-Variablen **UTC_OFFSET** und **DAY_LIGHT** gesetzt sein.

Folgende Tasks stehen zur Verfügung:

Task	Funktionalität
NTP_SRV	Starten des NTP-Server
NTP_SRV_STOP	Stoppen des NTP-Server

Vorbedingung zur Nutzung des NTP-Servers ist eine geladene NTP-Library, da der Status des Servers (Stratum=2 oder 13) je nach Synchronisation des Rechners gemeldet wird und dazu wird eine Funktion aus der NTP-Library genutzt, die den Status der NTP-Synchronisation dieses Rechner feststellt.

6 Automatische Sommer/Winterzeit Umstellung

Zur automatischen Sommer/Winterzeit Umschaltung steht ein Tool (DAYLIGHT . SR) unter RTOS-UH zur Verfügung. Es müssen die ENV-Variablen **UTC_OFFSET** und **DAY_LIGHT** gesetzt sein.

Folgende Task steht für den geordneten Systemstart zur Verfügung :

Task	Funktionalität
INIT_DAYLIGHT	Setzen der Sommer/Winterzeit auf Grund des aktuellen Datums und lokalen Uhrzeit

Die NTP-Library muß geladen sein, da Funktionen aus ihr genutzt werden und bei Umstellungen der Sommer/Winterzeit die Tasks des NTP-Client genutzt werden. Die Umstellung setzt die Environment-Variable **DAY_LIGHT** entsprechend neu, so dass mit **NTP_DAY_LIGHT** der Zeitbereich abgefragt werden kann.

6.1 CHECK AND SET SUMMERTIME

Die Funktion **CHECK_AND_SET_SUMMERTIME** liest die Environmentvariable **DAY_LIGHT** aus und überprüft an Hand des aktuellen Datums und Uhrzeit ob eine Umstellung der Zeit notwendig ist. Sie liefert den Wert **TRUE ('1'B)** zurück, wenn eine Zeitumstellung vorgenommen wurde, damit der Anwender im eigenen Programm geeignete Maßnahmen durchführen kann. Die Funktion muss von einer Task des Anwenders zyklisch aufgerufen werden, damit sie die Überprüfung vornehmen kann.

SPC CHECK_AND_SET_SUMMERTIME RETURNS(BIT(1)) GLOBAL ;

6.2 GET AKT WOCHENTAG

Die Funktion **GET_AKT_WOCHENTAG** gibt den aktuellen Wochentag zurück.

SPC CHECK_AND_SET_SUMMERTIME RETURNS(FIXED) GLOBAL ;

Die Wochentag sind folgend durchnummeriert:

Wochentag	Sonntag	Montag	Dienstag	Mittwoch	Donnerstag	Freitag	Samstag
Wert	0	1	2	3	4	5	6

6.3 Beispiel Zeitumstellung

```
SPC CHECK_AND_SET_SUMMERTIME RETURNS( BIT(1)) GLOBAL ;
...
MEINE_ZEITUMSTELLUNG:TASK ;
  DCL change BIT(1) ;
  DCL is_sommer FIXED(31) ;
  change = CHECK_AND_SET_SUMMERTIME ;
  IF change THEN
    /* Uhr ist umgestellt worden */
    ...
    /* Reaktion der eigenen Anwendung .... */
    ...
  FIN ;
  is_sommer = NTP_DAY_LIGHT ;
  IF is_sommer EQ 1(31) THEN /* Sommerzeit */
    AT 3:00:00 ALL 24 HRS ACTIVATE MEINE_ZEITUMSTELLUNG ;
  ELSE /* Winterzeit */
    AT 2:00:00 ALL 24 HRS ACTIVATE MEINE_ZEITUMSTELLUNG ;
  FIN ;
END ;
```