

6. Zeitglieder und Zähler



-> hierzu Folgen 6 und 7 der praktischen Einführung in CoDeSys

6.1 Zeitglieder (Timer) nach IEC 61131-3

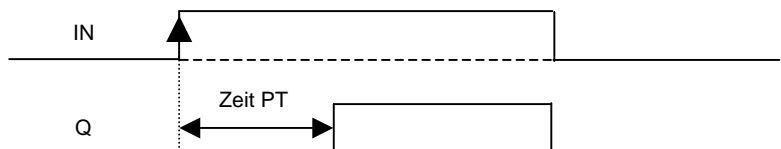
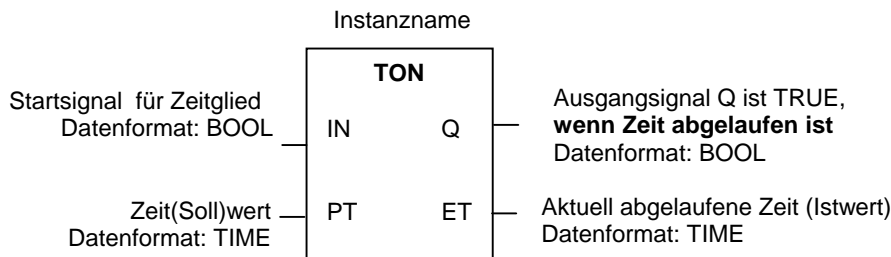
Die Norm IEC 61131-3 kennt drei Typen von Zeitgliedern (Timer):

- **Einschaltverzögerung** **TON** **(Timer - ON delay)**
- **Ausschaltverzögerung** **TOF** **(Timer – OFF delay)**
- **Impulsgeber** **TP** **(Timer Pulse)**

Wie R-S-FlipFlop und Trigger sind auch Timer **parametrierbare Funktionsblöcke** der Standard-Bibliothek und als solche zu instanzieren (**Bild 6-1**).

Nachfolgend sind die Parameter und die grundsätzliche Wirkungsweise der drei Timertypen aufgeführt. Die Kürzel der Parameter stehen für

IN	Input	Startbit	Datentyp: BOOL
Q	Quit	Zeitbit (Sollwert erreicht)	Datentyp: BOOL
PT	Preset Time	Sollwert der Zeitverzögerung	Datentyp: TIME
ET	Elapse Time	Istwert der Zeitverzögerung	Datentyp: TIME



Beispiel in AWL:

```

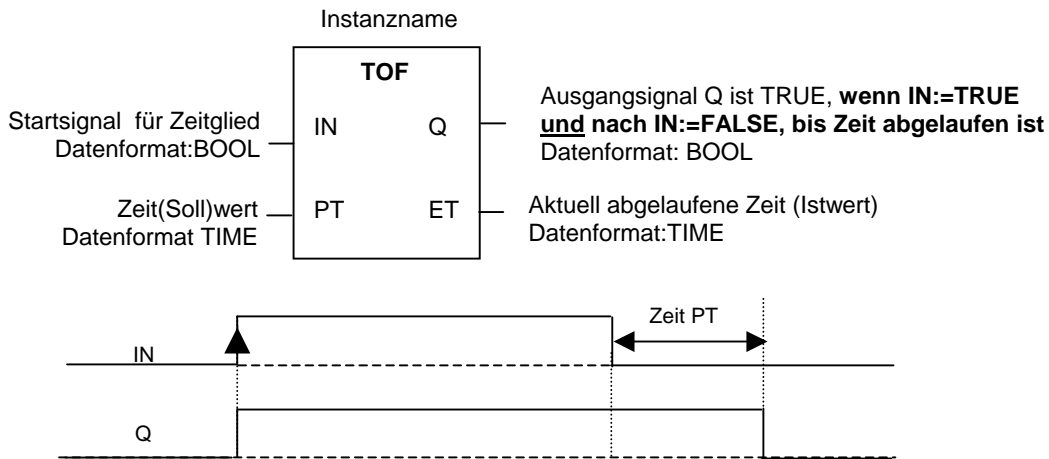
VAR
Pausenzeit:TON;
Start:BOOL;
Ventil_1:BOOL;
END_VAR

LD Start
ST Pausenzeit.IN
CAL Pausenzeit (PT:= T#2min5s)
LD Pausenzeit.Q
S Ventil_1
    
```

Beim Aufruf der Instanz mit dem Befehl CAL können INPUT- Parameter übergeben werden.

OUTPUT - Parameter müssen dagegen im Programm unabhängig vom Aufruf behandelt werden.

Es ist auch möglich, einzelne oder alle INPUT-Parameter vor dem Aufruf zuzuweisen.

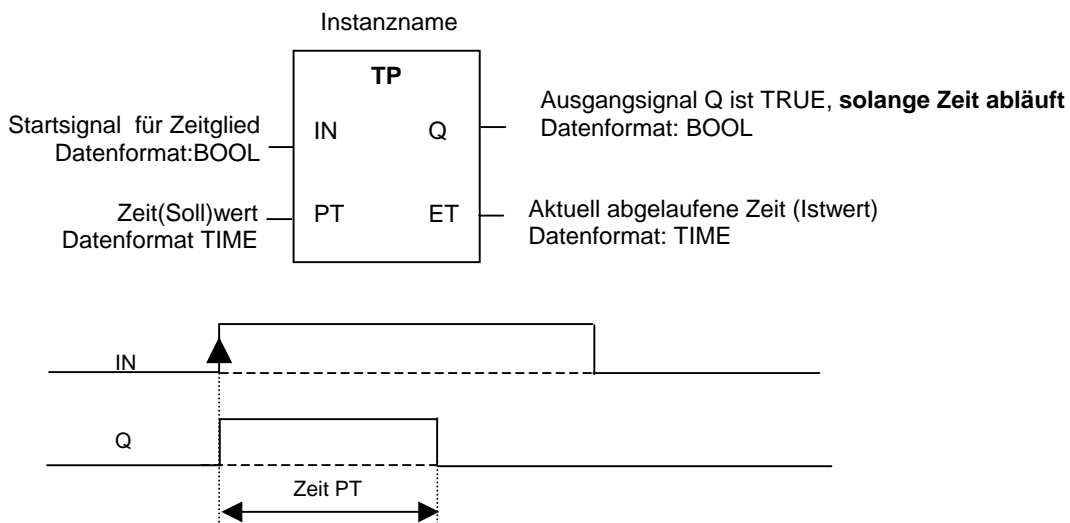


Beispiel in AWL:

```

VAR
Nachlauf: TOF;
Maschine_Start: BOOL;
Antrieb: BOOL;
END_VAR

CAL Nachlauf (IN:= Maschine_Start, PT:= T#5s)
LD Nachlauf.Q
ST Antrieb
    
```



Beispiel in AWL:

```

VAR
Impuls: TP;
Start: BOOL;
Stopp: BOOL;
Restzeit: TIME;
END_VAR

CAL Impuls (IN:=Start, PT:=T#5s)
LD Impuls.Q
ST Stopp
LD Impuls.ET
ST Restzeit
    
```

Das internationale **Datenformat TIME** steht für eine Zeitdauer und wird in Tagen, Stunden, Minuten, Sekunden und Millisekunden angegeben.

Datentyp	Schlüsselwort	Innere Wertung	Beispiel
Zeitdauer	TIME	Zeitdauer in Millisekunden	T#2h30ms

Die Angaben Tag-Std-Min-Sek-Millisek sind immer der Größe nach zu ordnen, aber nicht alle Angaben sind zu besetzen.

Beispiele: Zeitwert_1:=T#20ms; Zeitwert_2:=T#2min10ms; Zeitwert_3:=T#2d4h3s10ms;

Zeitbasis ist die **Millisekunde**, d.h. intern abgespeichert wird eine Anzahl Millisekunden in ein Doppelwort. Damit sind Konvertierungen in andere Datenformate problemlos möglich. Der größtmögliche Dezimalwert eines Doppelwortes ist 4.294.967.295. Dies ist deshalb die maximal mögliche Zeitvorgabe in Millisekunden. Ein Tag hat beispielsweise 86.400.000 ms.



Vor Auswahl der Typen TON, TOF oder TP stets sorgfältig überlegen!

90 % aller Zeitprobleme können allein mit der Einschaltverzögerung TON gelöst werden.

TON wird eingesetzt, wenn man zeitverzögert zu einem Signalwechsel ein zweites Signal benötigt, oder wenn man die Wirkung eines Signalwechsels verzögern will.

TP wird eingesetzt, wenn man ein Signal einer bestimmten Zeitdauer benötigt.

TOF wird eingesetzt, wenn man die Wirkung eines Signalwechsels verzögern will.

6.2 Zähler (Counter) nach IEC 61131-3

In der Norm IEC 61131-3 werden drei Typen von Zählern aufgeführt:

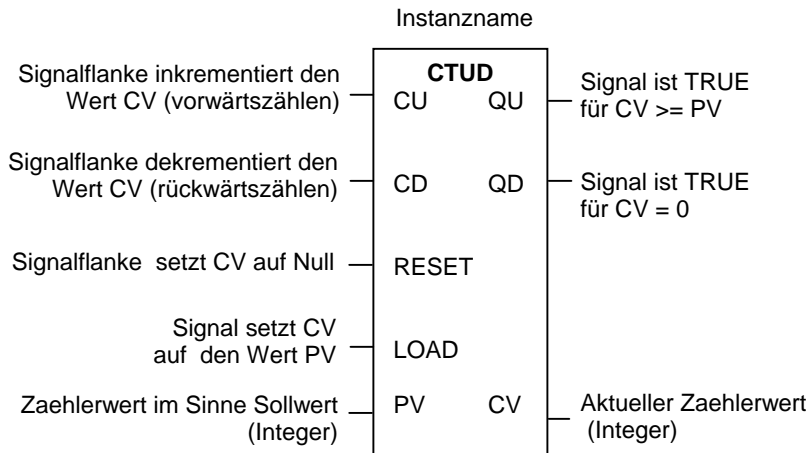
- **Vorwärtszähler** **CTU** **(Count up)**
- **Rückwärtszähler** **CTD** **(Count down)**
- **Vor- und Rückwärtszähler** **CTUD** **(Count up and down)**

Wie R-S-FlipFlop, Trigger und Timer sind auch Zähler **parametrierbare Funktionsblöcke** der Standard-Bibliothek und als solche zu instanzieren (**Bild 6-1**).

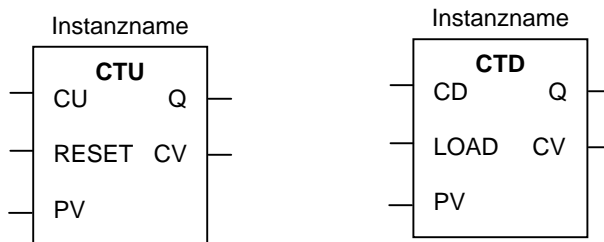
Nachfolgend sind die Parameter und die grundsätzliche Wirkungsweise der drei Typen von Countern aufgeführt. Die Kürzel der Parameter stehen für

CU	Count Up	Aufwärtszählen	Datentyp:BOOL
CD	Count Down	Abwärtszählen	Datentyp:BOOL
RESET	Reset	Rücksetzen	Datentyp:BOOL
LD	Load	(Wert) Laden	Datentyp:BOOL
Q	Quit	Ausgang (Sollwert erreicht)	Datentyp:BOOL
QU	Quit Up	Sollwert erreicht bei CU	Datentyp:BOOL
QD	Quit Down	Sollwerterreicht bei CD	Datentyp:BOOL
PV	Preset Value	Sollwert	Datentyp:INT
CV	Current Value	Istwert, aktueller Wert	Datentyp:INT

In Folge der Festlegung des Datentyps für CV als Integer mit 16 Bit Breite liegt der Zählbereich mit 0..32 767 fest. Das höchstwertige Bit ist für das Vorzeichen des INT reserviert. Nach Erreichen der Grenzen 32767 bzw. 0 reagiert der Zähler auf weitere Signale an den Eingängen CU und CD nicht.



Die Zaehler CTU und CTD sind vereinfachte Varianten des CTUD. Sie besitzen jeweils nur ihren speziellen Zaehleinang CU oder CD. Typ CTU erlaubt das Rücksetzen auf Null, CTD dagegen das Laden eines (Soll)Wertes. Die Ausgänge Q übernehmen jeweils die Funktion der Ausgänge QU bzw. QD.



Beispiel Rückwärtszähler in AWL:

```

VAR
Teilezaehler:CTD;
LS_Gutteile:BOOL;
Sollwert_setzen:BOOL;
Sollwert:INT:=1000;
Ventil_1:BOOL;
Anzeige:INT;
END_VAR
    
```

```

CAL Teilezaehler (CD:=LS_Gutteile, LOAD:=Sollwert_setzen, PV:=Sollwert )
LD Teilezaehler.Q
R Ventil_1
LD Teilezaehler.CV
ST Anzeige
    
```

Beim Aufruf der Instanz mit dem Befehl CAL können INPUT- Parameter übergeben werden.

OUTPUT - Parameter müssen dagegen im Programm unabhängig vom Aufruf behandelt werden.

Es ist auch möglich, einzelne oder alle INPUT-Parameter vor dem Aufruf zuzuweisen.

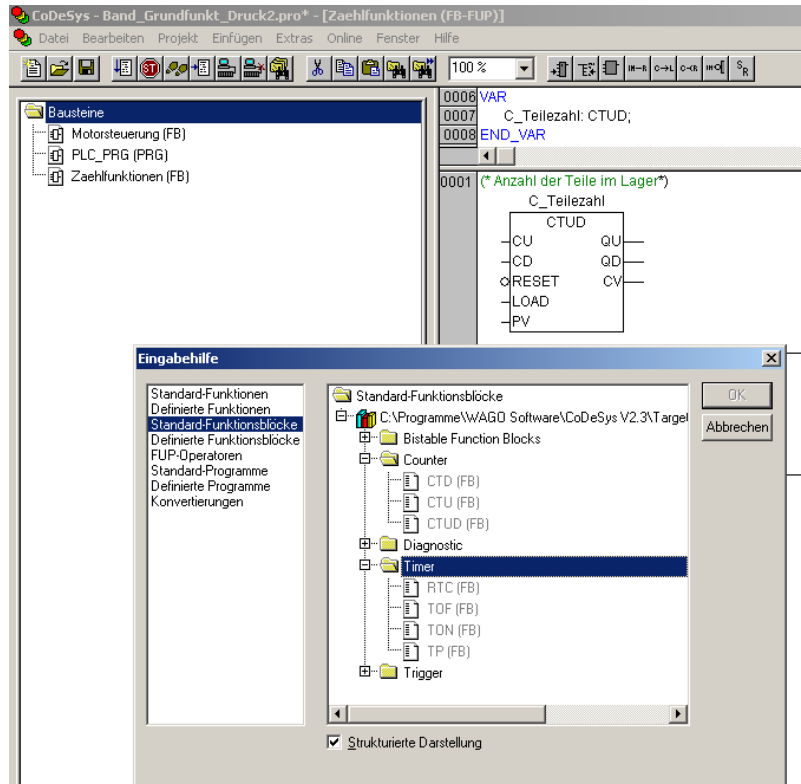


Bild 6-1: Verfügbare Standardfunktionsblöcke Counter und Timer in CoDeSys. Oben wurde ein CTUD in das Programm eingefügt und instanziiert.

6.3 Zum Vergleich: Timer im System Step7

In Step7 kann man traditionell **5 Typen von Zeitgliedern** verwenden:

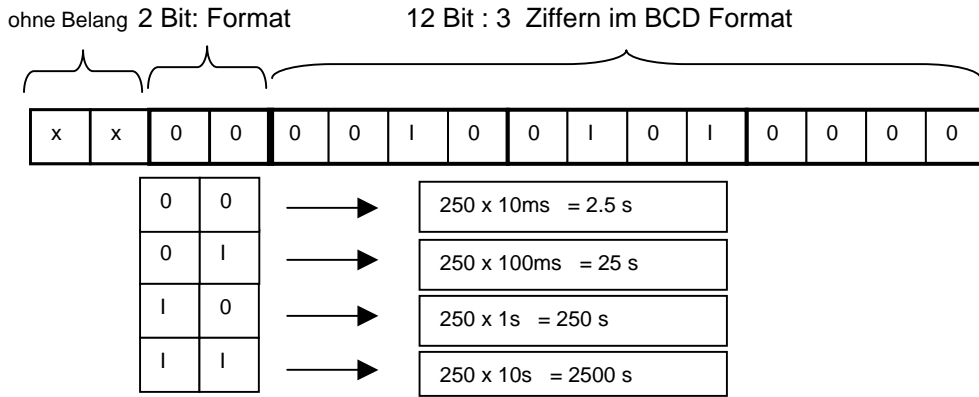
- Einschaltverzögerung SE
- Speichernde Einschaltverzögerung SS
- Impuls SI
- Verlängerter Impuls SV
- Ausschaltverzögerung SA

Die Typen SS und SV haben hinsichtlich ihrer Wirkung intern einen Speicher und können deshalb trotz Abbruch des Startsignals die vollständige Zeit herunterzählen. Der Typ SS muss deshalb fast immer zurückgesetzt werden.

Diese bewusst weiter als S5-Timer bezeichneten Zeitglieder sind nicht IEC-gerecht. Sie wurden bereits mit Simatic S5 / Step5 definiert und in Step7 übernommen. In Step7 werden die Zeitwerte im speziellen nicht genormten Zeitformat S5T#... unverschlüsselt vorgegeben. Gegenüber der Verschlüsselung der Zeitwerte bei Step5 (z.B. in Form KT 100.2 für 100s) erleichtert dies die Programmierung.

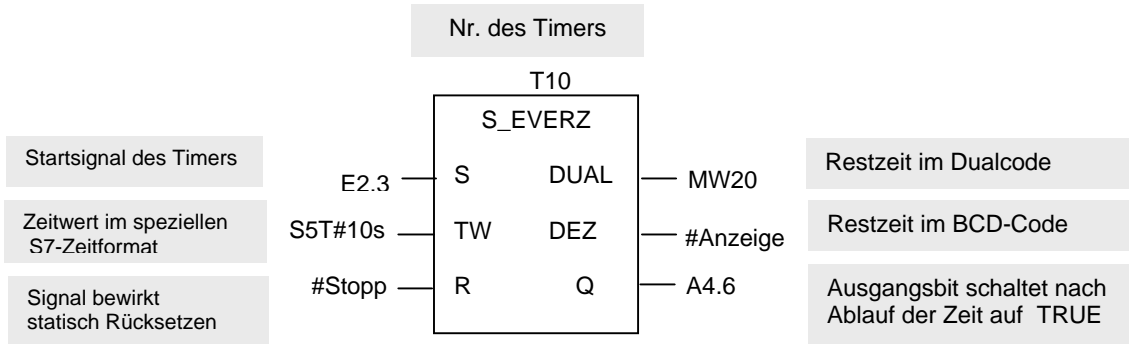
Eine S7-CPU kann je nach Typ und Leistungsfähigkeit eine bestimmte Anzahl Timer verwalten, so z.B. die CPU 315-2DP/PN 128 Timer T0 ..T127. Jedem dieser Timer ist intern ein Zeitwort mit nachfolgendem Inhalt zugeordnet.

- **Zeitwort von S5-Timern:**



Mit dieser Festlegung können mit S5-Timern Zeiten von 10ms .. 9990s bearbeitet werden.

Nachfolgendes Schema sowie **Bild 6-2** verdeutlichen die anzuschaltenden Signale und ihre Wirkungen. Typisch für Step7 wurden im Beispiel absolute Adressen verwendet, aber auch zwei lokale temporäre Variablen. In AWL wurde mit symbolischen Operanden gearbeitet, wobei die Symboltabelle zwischen absoluten und symbolischen Operanden vermittelt.



U	"Start"
L	S5T#10s
S	"Zeitglied_1"
U	#Stopp
R	"Zeitglied_1"
L	"Zeitglied_1"
T	"Zwischenspeicher"
LC	"Zeitglied_1"
T	#Anzeige
U	"Zeitglied_1"
=	"Ventil_1"

Operand	Symbol
E2.3	Start
MW 20	Zwischenspeicher
A4.6	Ventil_1
T10	Zeitglied_1

ang\S7-Programm(3)\...\FC2]

Ansicht Extras Fenster Hilfe

Inhalt von: 'Umgebung\Schnittstelle\TEMP'

Name	Datentyp	Adresse	Kommentar
Stopp	Bool	0.0	
Anzeige	Word	2.0	

Netzwerk 2: Wartezeit

Kommentar:

T10 "Zeitglied" \$_EVERZ

E2.3 "Start" S DUAL "Zwischensp-eicher" MW20

S5T#10S TW DEZ #Anzeige A4.6 "Ventil_1"

#Stopp R Q =

Bild 6-2: AWL und FUP eines S5-Timers des Typs Einschaltverzögerung bei Verwendung von symbolischen Operanden und lokalen temporären Variablen

Bild 6-2 zeigt rechts einen Ausschnitt des Programms in Sprache FUP. Der Timer wurde in der Funktion FC2 unter Verwendung der beiden temporärer Variablen # Stopp vom Typ BOOL und #Zwischenspeicher vom Typ WORD programmiert.

6.4 Zum Vergleich: Zähler im System Step7

In Step7 kann man traditionell die gleichen **3 Typen von Zählern** verwenden. Die Typen

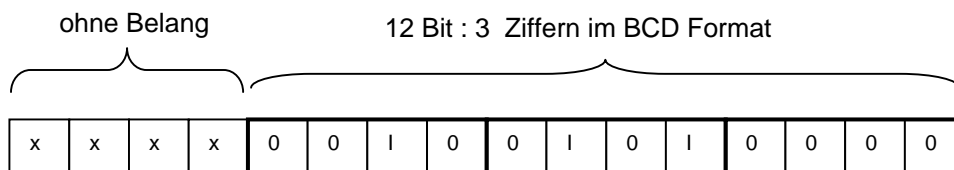
- Vorwärtszähler
- Rückwärtszähler
- Vor- und Rückwärtszähler

haben grundsätzlich gleiche Funktionen, jedoch nicht die gleichen Zählgänge.

Die speziellen S5-Zähler sind nicht IEC-gerecht. Sie wurden bereits mit Simatic S5 / Step5 definiert und in Step7 übernommen. Sollen die Zähler auf einen Wert gesetzt werden, so ist dafür in Step7 das nicht genormte Format C#... zu benutzen.

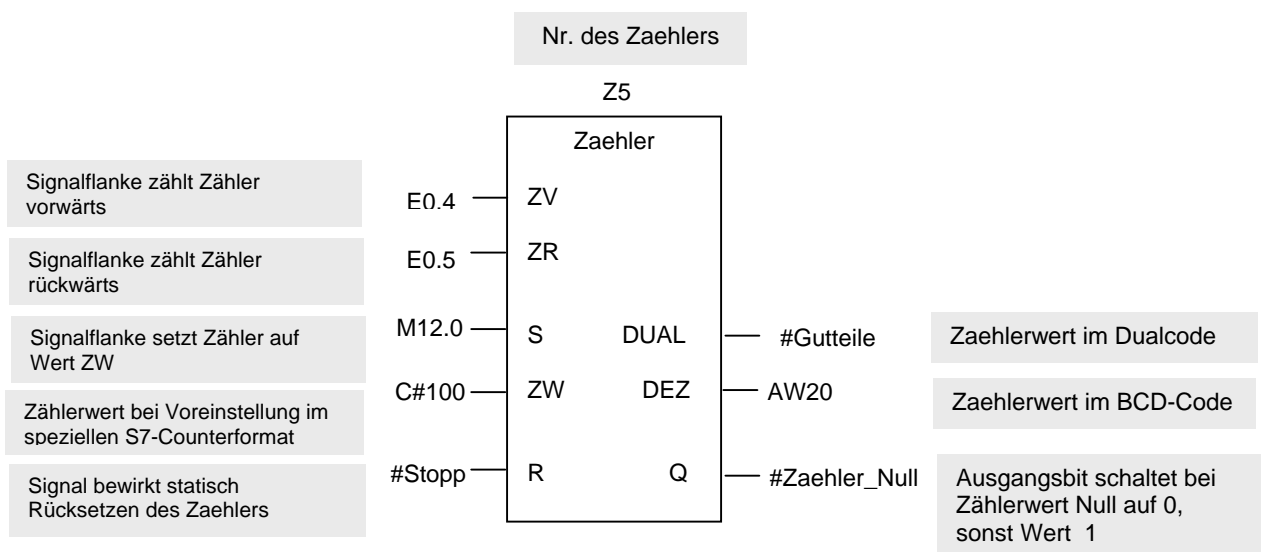
Eine S7-CPU kann je nach Typ und Leistungsfähigkeit eine bestimmte Anzahl Zähler verwalten, so z.B. die CPU 315-2DP/PN 128 Zähler Z0 ..Z127. Jedem dieser Zähler ist intern ein Zählerwort mit nachfolgendem Inhalt zugeordnet.

- **Zählerwort von S5-Zählern:** Zaehlerinhalt 250



Mit dieser Festlegung können S5-Zähler im Bereich 0 .. 999 zählen.

Nachfolgendes Schema und **Bild 6-3** verdeutlichen die anzuschaltenden Signale und ihre Wirkungen. Typisch für Step7 wurden im Beispiel absolute Adressen verwendet, aber auch lokale temporäre und statische Variablen. In AWL wurde mit symbolischen Operanden gearbeitet, wobei die Symboltabelle zwischen absoluten und symbolischen Operanden vermittelt.



Der Zähler Z5 im Bild 6-3 wurde in einem Funktionsbaustein FB2 unter Verwendung der statischen Variablen #Gutteile vom Typ INT und #Zaehler_Null vom Typ BOOL sowie der temporären Variablen #Stopp vom Typ BOOL programmiert.

U	"LS-Fehlerteile"
ZV	"Teilezaehler"
U	"LS_Gutteile"
ZR	"Teilezaehler"
U	"Zaehler_setzen"
L	C#100
S	"Teilezaehler"
U	#Stopp
R	"Teilezaehler"
L	"Teilezaehler"
T	#Gutteile
LC	"Teilezaehler"
T	"Anzeige_Gutteile"
U	"Teilezaehler"
=	#Zaehler_Null

Operand	Symbol
E0.4	LS_Fehlerteile
E0.5	LS_Gutteile
M12.0	Zaehler_setzen
AW20	Anzeige_Gutteile
Z5	Teilezähler

The screenshot shows the SIMATIC Manager interface. The top part displays a variable declaration table for the 'Schmittstelle' object:

Name	Datentyp	Adresse	Anfangswert	Aussch
Gutteile	Int	0.0	0	
Zaehler_Null	Bool	2.0	FALSE	

The bottom part shows the AWL/FUP ladder logic for the 'Teilezaehler' function block. The logic includes the following rungs:

- Rung 1: LS-Fehlerteile (E0.4) normally open contact, ZV (Zähler setzen) coil.
- Rung 2: LS-Gutteile (E0.5) normally open contact, ZR (Zähler setzen) coil.
- Rung 3: Zaehler_setzen (M12.0) normally open contact, S (Setzen) coil.
- Rung 4: C#100 (Timer) normally open contact, ZW (Zählerwert) coil.
- Rung 5: #Stopp (U) normally open contact, R (Resets) coil.

Additional logic elements include a DUAL coil for #Gutteile, a DEZ coil for 'Anzeige_Gutteile', and a coil for #Zaehler_Null.

Bild 6-3: AWL und FUP eines S5-Vor-Rückwärtszählers mit symbolischen Operanden und lokalen statischen und temporären Variablen

6.5 Benutzung von IEC-Timern und IEC-Countern in Step7

Es ist im System Simatic S7 / Step7 auch möglich, Timer und Counter nach IEC 61131-3 zu verwenden. Dazu werden System Funktionsblöcke der Standard-Bibliothek benutzt (**Bild 6-4 links**).

- Timer der Typen TP, TOF und TON stehen als SFB 3, SFB 4 und SFB 5 zur Verfügung
- Counter der Typen CTU, CTD und CTUD stehen als SFB 0, SFB 1 und SFB 2 zur Verfügung

Abweichend von der Instanziierung solcher Funktionsblock nach IEC 61131-3 werden in Step7 solche Blocks mit Instanzdatenbausteinen aufgerufen (hierzu siehe auch Abschnitt 9). **Bild 6-4 rechts** zeigt die Aufrufe in FUP, **Bild 6-5** in AWL.

Die Instanzdatenbausteine DB10 bzw. DB11 für die Aufrufe zeigt **Bild 6-6**. Nochmals sei bemerkt, dass diese DB vom System selbsttätig generiert werden. In den Instanzdatenbausteinen erscheinen alle Parameter der Typen IN und OUT sowie einige für die Funktion erforderliche statische Variablen, vergleichbar mit Zeit- und Zählerworten von Step7.

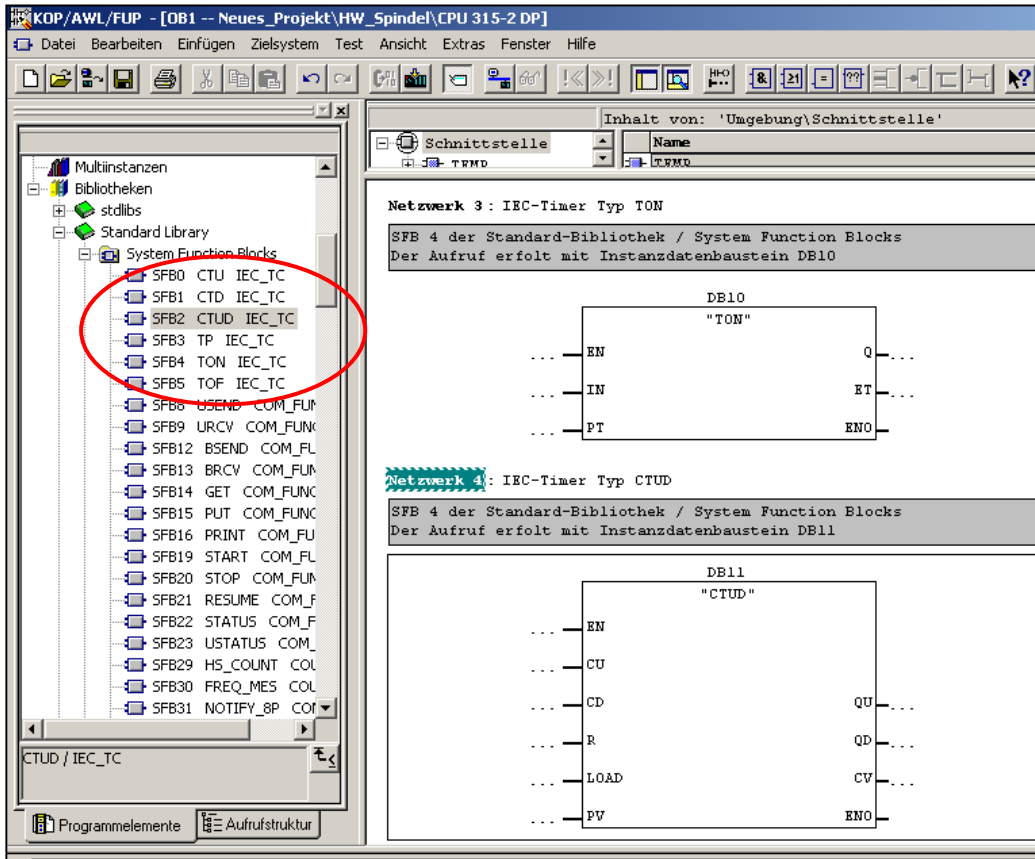


Bild 6-4: Einfügen von IEC-Timern und –Zählern im System Step7. Darstellung FUP

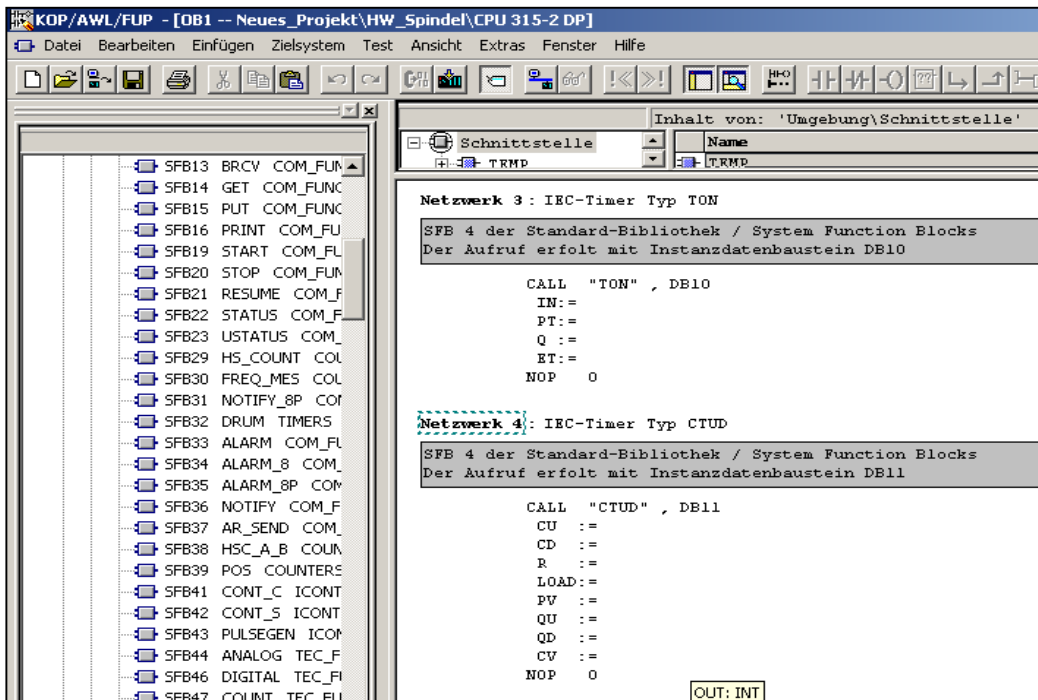


Bild 6-5: Einfügen von IEC-Timern und –Zählern im System Step7. Darstellung AWL

	Adresse	Deklaration	Name	Typ	Anfangswert	Aktualwert	Kommentar
1	0.0	in	IN	BOOL	FALSE	FALSE	
2	2.0	in	PT	TIME	T#0MS	T#0MS	
3	6.0	out	Q	BOOL	FALSE	FALSE	
4	8.0	out	ET	TIME	T#0MS	T#0MS	
5	12.0	stat	STATE	BYTE	B#16#0	B#16#0	
6	14.0	stat	STIME	TIME	T#0MS	T#0MS	
7	18.0	stat	ATIME	TIME	T#0MS	T#0MS	

	Adresse	Deklaration	Name	Typ	Anfangswert	Aktualwert	Kommentar
1	0.0	in	CU	BOOL	FALSE	FALSE	
2	0.1	in	CD	BOOL	FALSE	FALSE	
3	0.2	in	R	BOOL	FALSE	FALSE	
4	0.3	in	LOAD	BOOL	FALSE	FALSE	
5	2.0	in	PV	INT	0	0	
6	4.0	out	QU	BOOL	FALSE	FALSE	
7	4.1	out	QD	BOOL	FALSE	FALSE	
8	6.0	out	CV	INT	0	0	
9	8.0	stat	CUO	BOOL	FALSE	FALSE	
10	8.1	stat	CDO	BOOL	FALSE	FALSE	

Bild 6-6: Instanzdatenbaustein von IEC-Timer TON (oben) und CTUD (unten) gemäß Aufruf nach nach Bild 6-4. Diese Instanzdatenbausteine ersetzen bei Step 7 die Instanzen.

- **Hinweis:**

Einen besonderen Fall stellt das **System Step7MicroWin32 für S7-200** dar: Hier kann man sich entscheiden, ob gemäss Simatic-Regeln oder nach dem Standard IEC 61131-3 programmiert werden soll (Einstellung erfolgt unter -> *Extras* -> *Optionen*). Bei einer Entscheidung für Simatic kann man darüber hinaus die Sprache „*Simatic*“ oder „*International*“ wählen.

Wählt man Spracheinstellung „*Simatic*“, so stehen IEC-ähnliche Timer der Typen TON und TOF und darüber hinaus auch die nachtriggerbare Einschaltverzögerung TONR zur Verfügung. Allerdings gibt man bei diesen Timern den Zeitwert PT nicht mit dem Datentype TIME vor, sondern schreibt einen Integer-Wert an. Zeitbasis ist ein Zählschritt. Man verwendet bis zu 32 767 Zählschritte mit drei möglichen Zeitauflösungen. Diese sind mit der Nummer des Timers fest vorgegeben. So werden eine bestimmte Zahl Timer mit jeweils vorgegebener Auflösung definiert. Den Ausgangsparameter ET für die aktuelle Zeit gibt es bei diesen Timern nicht.

Auflösung (Zählschritt)	1 ms	10 ms	100 ms
Maximaler Zeitwert	32,767 s	327, 67 s	3276,7 s
Typ der Timer	TON, TOF, TP (Einstellung Sprache IEC)		
Verwendbare Timer	T32, T96	T33-T36, T97-T100	T37-T63, T101-T255
Typ der Timer	TONR (nur bei Einstellung Sprache Simatic)		
Verwendbare Timer	T0, T64	T1-T4, T65-T68	T5-T31, T69-T95

Wählt man Spracheinstellung „*IEC 61131*“, so stehen Timer vom Typ TON, TOF oder TP mit Ausgangsparameter ET zu Verfügung. Allerdings muss man auch hier die Zeitwerte weiter mit Zählschritten bis 32 767 vorgeben. Die Ausgangswerte des Parameters ET haben folgerichtig Wort- und Doppelwortformat. Details werden immer wieder unterschiedlich gehandhabt. Bei Nutzung der Online- und Kontextsensitiven Hilfe stellt dies aber kein wirkliches Problem dar, allerdings bleiben manche Programme damit herstellerspezifisch!