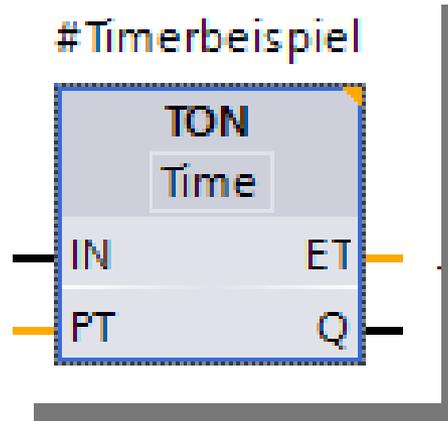




IEC - Zeitfunktionen





Zeitfunktionen nach IEC 61131-3

Zeitfunktionen (Timer) in einer SPS

In SPS-Programmen werden häufig Zeitfunktionen benötigt, um z.B. eine zeitgesteuerte Aktion auszulösen. Das kann z.B. ein Zylinder sein, der in Endlage mehrere Sekunden verbleiben soll, ehe er wieder zurück bewegt wird. Oder eine Alarmleuchte, die für mehrere Sekunden aufleuchten soll. Für solche Aktionen benötigt man Zeitglieder, mit denen zeitgesteuerte Funktionen programmiert werden können.

Gemäß der **Programmierenorm 61131-3** wurden **drei Zeitfunktionen** festgelegt:

Funktionsbaustein	Impuls	TP	Timer Puls
Funktionsbaustein	Einschaltverzögerung	TON	Timer ON-Delay
Funktionsbaustein	Ausschaltverzögerung	TOF	Timer OFF-Delay



Zeitfunktionen nach IEC 61131-3

Impulsdiagramme zu den drei Zeitgliedern der IEC 61131-3

Name	Funktion Grafische Darstellung	Zeitdiagramm
TP	Erzeugen eines Impulses 	
TON	Einschaltverzögerung 	
TOF	Ausschaltverzögerung 	

Anweisungen

Optionen

Favoriten

Einfache Anweisungen

Name	Beschreibung
IEC-Zeiten	
TP	Impuls erzeugen
TON	Einschaltverzögerung erzeugen
TOF	Ausschaltverzögerung erzeugen

Hinweis:
In STEP7 zu finden in der Bibliothek

- Bibliotheken
 - stdlibs
 - Standard Library
 - Miscellaneous Blocks
 - T1-S7 Converting Blocks
 - PID Control Blocks
 - IEC Function Blocks
 - System Function Blocks
 - SFB0 CTU IEC_TC
 - SFB1 CTD IEC_TC
 - SFB2 CTUD IEC_TC
 - SFB3 TP IEC_TC
 - SFB4 TON IEC_TC
 - SFB5 TOF IEC_TC



Zeitfunktionen nach IEC 61131-3

Beispiel: Einschaltverzögerter **TON**

TON = Timer On delay = einschaltverzögert

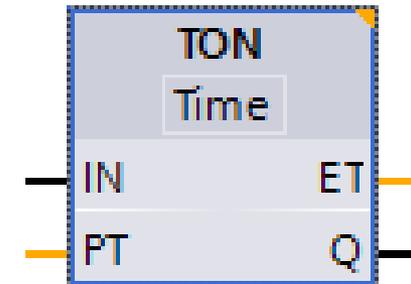
Beschreibung der Ein- und Ausgangsvariablen:

IN	BOOL	Start des Timers (nicht nachtriggerbar)
PT	TIME	Vorgabezeitwert (preset time) Zeitwert, z.B.: T#3s
ET	TIME	abgelaufene Zeit in ms (elapsed time)
Q	BOOL	Q=1, wenn die Zeit abgelaufen und solange IN=1

Format des Datentyps **TIME**:

- Ganzzahl 32 Bit in ms
- Der Zeitwert kann in d,h,m,s,ms angegeben werden
- Bsp: T#5d3h2m4s8ms (Tage, Stunden, Minuten, Sekunden, Millisekunden)
- Bereich: T#0s ... T# 24D 20h 31m 23s 647ms

#Timerbeispiel

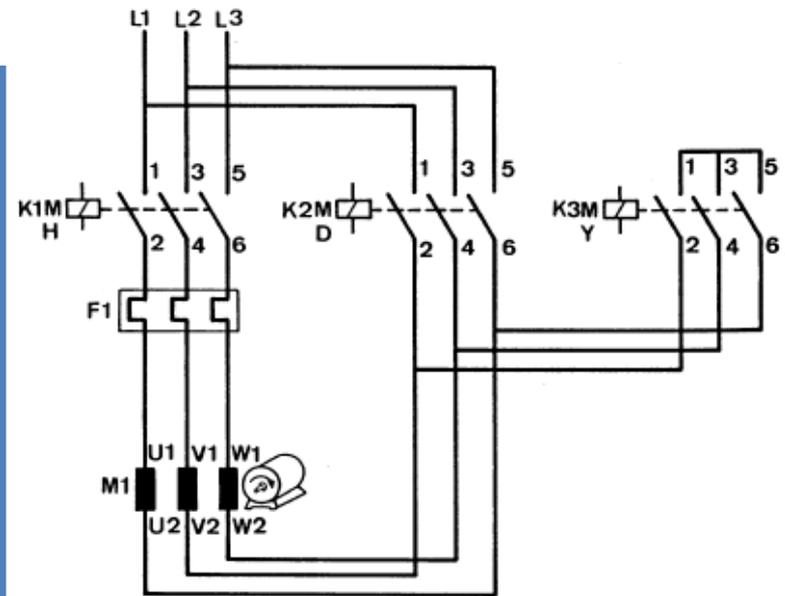




Lernaufgabe: „Stern-Dreieck-Anlauf“

Für einen automatischen Stern-Dreieck-Anlauf eines Drehstrommotors soll ein Baustein mit folgender Funktionalität entwickelt werden:

- Wird der Taster S1 betätigt, so zieht das Netzschütz K1M an und bleibt geschaltet.
- Gleichzeitig mit K1M soll auch das Sternschütz K3M anziehen.
- Nach einer parametrierbaren Verzögerungszeit soll das Sternschütz K3M abgeschaltet und das Dreieckschütz K2M eingeschaltet werden.
- Wird der Taster S0 betätigt, so fallen alle aktuell betätigten Schütze ab.



1. Bestimmen Sie die **Zuordnungstabelle** der Ein- und Ausgänge der SPS und legen Sie dabei die richtige Zuordnung der Signalgeber (Öffner, Schließer) fest.
2. Deklarieren Sie die benötigten **globalen Variablen** gemäß Zuordnungstabelle
3. Definieren Sie die **Bausteinschnittstelle** für den zu entwickelnden Baustein namens **SternDreieckAnlauf**. Als Standardumschaltzeit sollen 5s vorgesehen werden.
4. Deklarieren Sie die lokalen Bausteinvariablen und entwickeln Sie den normgerechten Baustein.
5. Rufen Sie den Baustein in **Main** (OB1) auf, beschalten Sie diesen gemäß Ihrer Zuordnungstabelle und testen Sie den Baustein ausgiebig.
6. Erstellen Sie eine eigene **Bibliothek** mit Ihrem Namen. Erstellen Sie darin einen Unterordner namens **Zeitfunktionen** und legen Sie diesen Baustein darin ab.

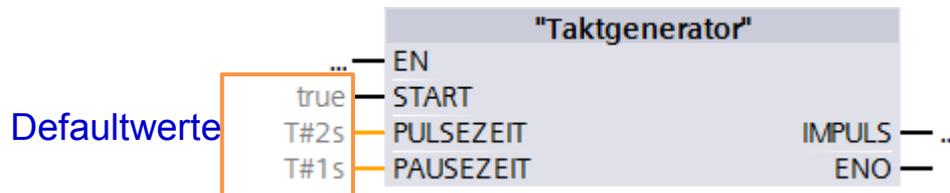


Lernaufgabe: „Taktgenerator“

Ein Taktgenerator mit **einstellbarem Puls-Pausenverhältnis** soll erstellt und in einer selbst zu erstellenden Bibliothek bereitgestellt werden. Wird der Anschluss „Start“ nicht belegt, soll der Generator ständig takten. Die Defaultwerte für die Pausezeit soll 1 Sekunde und für die Pulszeit soll 2 Sekunden betragen.

1. Definieren Sie die **Bausteinschnittstelle** für den zu entwickelnden Baustein namens **Taktgenerator**.
2. Deklarieren Sie die lokalen Bausteinvariablen inkl. den Defaultwerten
3. Programmieren Sie den normgerechten Baustein mit den entsprechenden IEC-Zeitfunktionen.
4. Rufen Sie den Baustein in **Main** auf, beschalten Sie diesen und testen Sie den Baustein ausgiebig.
5. Legen Sie diesen Baustein in Ihre Bibliothek ab.

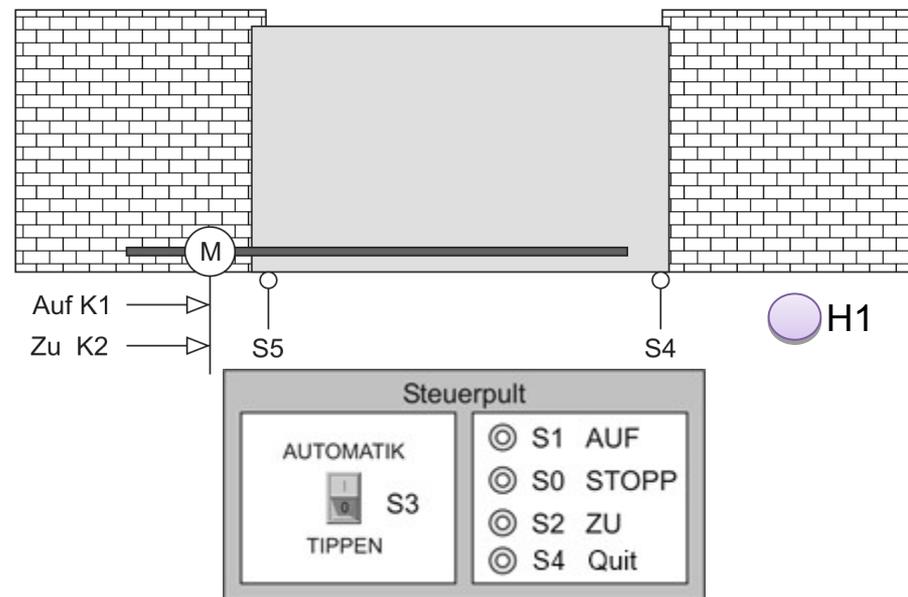
1. Bausteinschnittstelle





Lernaufgabe: „Fahrzeitüberwachung“

Die Torsteuerung eines Werktores soll um eine Fahrzeitüberwachung erweitert werden. Dazu muss die Fahrbewegung des Tores (Tor auf und Tor zu) im Automatikbetrieb zeitlich überwacht werden. Erreicht das Tor in einer bestimmten vorgegebenen Zeit die gegenüberliegende Endlage nicht, so liegt ein Fehler vor, der an einer Meldeleuchte H1 visualisiert werden soll. Die Ansteuerung des Tores über die Schütze Q1 bzw. Q2 muss abgeschaltet werden. Nachdem der Fehler über einen Quittiertaster S4 quittiert wurde, dürfen die Schütze erneut angesteuert werden um den Fahrvorgang voll zu beenden. Erreicht das Tor auch dann wieder die gegenüberliegende Endlage nicht, so muss der Fehler erneut ausgelöst werden.



1. Öffnen Sie den vorgegebenen Baustein **Tor** und veranschaulichen Sie sich die Ansteuerung.
2. Definieren Sie die neue **Bausteinschnittstelle** für den zu erweiternden Baustein **Tor** und deklarieren Sie die benötigten **globalen PLC-Variablen S4** (Quit) und **H1** (Meldeleuchte)
3. Programmieren Sie die geforderte Funktionserweiterung der Fahrzeitüberwachung mit einer geeigneten IEC-Zeitfunktion im Baustein **Tor**.
4. Rufen Sie den Baustein in **Main** auf, beschalten Sie diesen und testen Sie den Baustein ausgiebig.