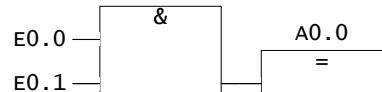
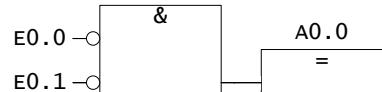
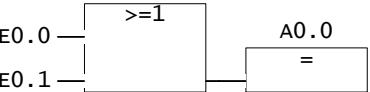
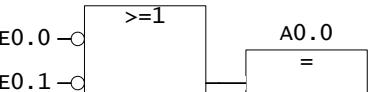
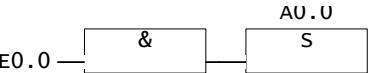
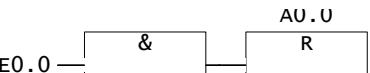


OB1 - <offline>**"Cycle Execution"**

Name: Familie
Autor: Version: 00.01
Bausteinversion: 2
Zeitstempel Code: 19.01.2002 11:58:51
Interface: 15.01.2002 22:22:24
Längen (Baustein / Code / Daten): 00512 00356 00020

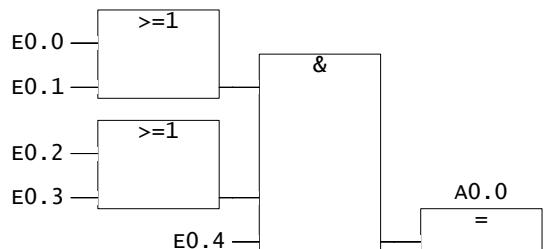
| Adresse | Deklaration | Name | Typ | Anfangswert | Kommentar |
|---------|-------------|---------|--------------|-------------|-----------|
| 0..0 | temp | Default | ARRAY[1..20] | | |
| *1..0 | temp | | BYTE | | |

Baustein: OB1 Grundschaltungen für die Formelsammlung**Netzwerk: 1 UND-Glied****Eingang E0.0 und Eingang E0.1 ist gleich Ausgang A0.0****Netzwerk: 2 UND-NICHT-Glied****Nicht Eingang E0.0 und nicht Eingang E0.1 ist gleich Ausgang A0.0****Netzwerk: 3 NICHT-Glied (in FUP nur mit UND)****Und nicht Eingang E0.0 ist gleich Ausgang A0.0****Netzwerk: 4 ODER-Glied****Eingang E0.0 oder Eingang E0.1 ist gleich Ausgang A0.0****Netzwerk: 5 ODER-NICHT-Glied****Nicht Eingang E0.0 oder nicht Eingang E0.1 ist gleich Ausgang A0.0****Netzwerk: 6 Setzen eines Ausgangs****Und Eingang E0.0 setzt Ausgang A0.0****Netzwerk: 7 Rücksetzen eines Ausgangs****Und Eingang E0.0 setzt Ausgang A0.0 zurück****Netzwerk: 8 NOP-Operation****Tue nichts (Nur in AWL)**

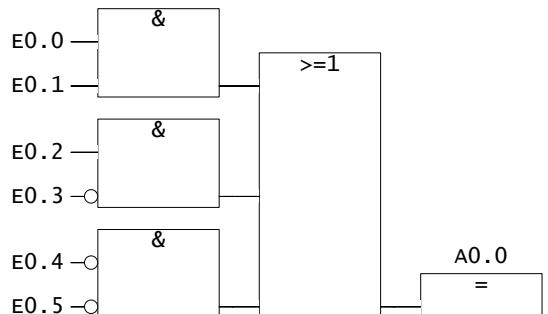
NOP 0 // Tue nichts (Nur in AWL)

Netzwerk: 9 UND-Verschachtelung

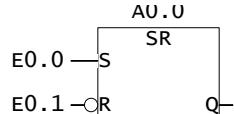
(Eingang E0.0 oder E0.1) und (Eingang E0.1 oder E0.3) und Eingang 0.4 ist gleich
Ausgang A0.0

**Netzwerk: 10 ODER-Verschachtelung**

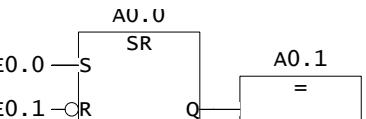
(Eingang E0.0 und Eingang E0.1) oder (Eingang E0.2 und nicht Eingang E0.3) oder
(Nicht Eingang E0.4 und nicht Eingang E0.5) ist gleich Ausgang A0.0

**Netzwerk: 11 Flip-Flop ohne Zustands-Auswertung**

Und Eingang E0.0 setzt Ausgang A0.0;
Und nicht Eingang E0.1 setzt Ausgang A0.0 zurück;
Rücksetzen hat Vorrang !!!

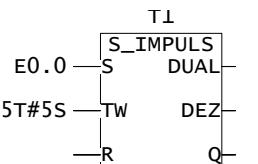
**Netzwerk: 12 Flip-Flop mit Ausgangsauswertung**

Und Eingang E0.0 setzt Ausgang A0.0;
Und nicht Eingang E0.1 setzt Ausgang A0.0 zurück;
Rücksetzen hat Vorrang !!!
Und Ausgang (des Flip-Flop) A0.0 ist gleich Ausgang A0.1

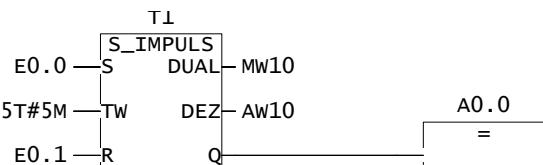
**Netzwerk: 13 Timer mit Impulsverhalten - Minimale AWL**

Und Eingang E0.0 lädt zeit (= 5 s) für Timer (Timerwort)
und startet Timer 1 mit Impulsverhalten.

Timer bleibt solange 1 bis E0.0 gleich 0 wird oder Zeit abgelaufen ist

**Netzwerk: 14 Timer mit Impulsverhalten - komplette AWL**

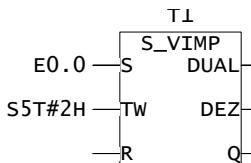
Und Eingang E0.0 lädt zeit für Timer (Timerwort, Zeit = 5 min)
und startet Timer 1 mit Impulsverhalten.
UND Eingang E0.1 setzt den Timer 1 zurück.
UND Timer 1 (Zeit läuft) ist gleich Ausgang A0.0



Netzwerk: 15 Timer mit verlängertem Impuls - Minimale AWL

Und Eingang E0.0 lädt Zeit (= 2 h) für Timer (Timerwort) und startet Timer 1 mit verlängertem Impuls.

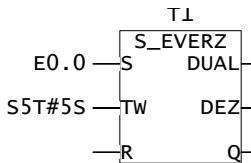
Timer bleibt solange 1 bis die Zeit abgelaufen ist.

**Netzwerk: 16 Timer mit Einschaltverzögerung - Minimale AWL**

Und Eingang E0.0 lädt Zeit (= 5 s) für Timer (Timerwort) und startet Timer 1 mit Einschaltverzögerung

Timer wird erst 1 wenn E0.0 noch gleich 1 ist und die Zeit abgelaufen ist.

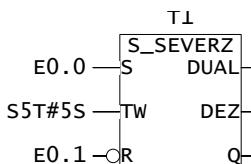
Timer wird 0 wenn E0.0 gleich 0 wird

**Netzwerk: 17 Timer mit speichernder Einschaltverzögerung - Minimale AWL**

Und Eingang E0.0 lädt Zeit (= 5 s) für Timer (Timerwort) und startet Timer 1 mit speichernder Einschaltverzögerung.
Und nicht Eingang E0.1 setzt den Timer 1 zurück.

Timer wird erst 1 wenn E0.0 noch gleich 1 ist und die Zeit abgelaufen ist.

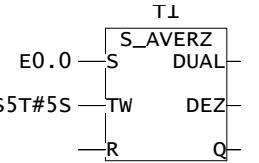
Timer bleibt auf 1 bis er zurückgesetzt wird.

**Netzwerk: 18 Timer mit Ausschaltverzögerung - Minimale AWL**

Und Eingang E0.0 lädt Zeit (= 5 s) für Timer (Timerwort) und startet Timer 1 mit Ausschaltverzögerung

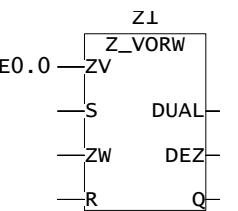
Timer wird 1 wenn E0.0 gleich 1 ist. Wenn E0.0 gleich 0 wird, startet die Zeit.

Timer bleibt 1 bis die Zeit abgelaufen ist

**Netzwerk: 19 Vorwärtzzähler - Minimale AWL**

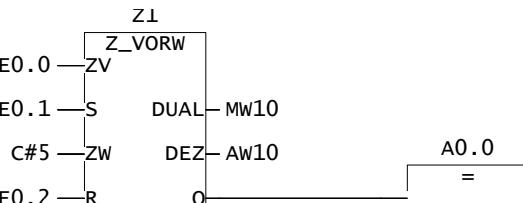
Und Eingang E0.0 zählt zähler 1 vorwärts.

Der Zähler zählt bis maximal 999 und zählt nicht weiter bis er zurückgesetzt wird.

**Netzwerk: 20 Vorwärtzzähler - Komplette AWL**

Und Eingang E0.0 zählt zähler 1 vorwärts.
Und Eingang E0.1 setzt zähler 1 auf bestimmten zählerstand
Und Eingang E0.2 setzt zähler 1 zurück
Und Zähler 1 (=Zustand) ist gleich Ausgang A0.0

Der Ausgang A0.0 ist 1 wenn Zähler 1 \neq 0 ist

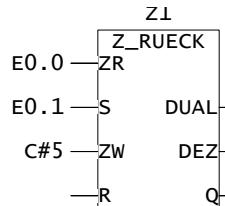


Netzwerk: 21 Rückwärtszähler - Minimale AWL

Und Eingang E0.0 zählt zähler 1 rückwärts.

Und Eingang E0.1 setzt zähler auf zählerstand 5

Der zähler zählt minimal bis 0

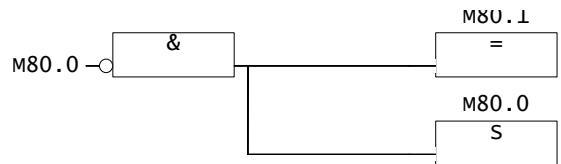


Netzwerk: 22 Richtimpuls

Und nicht Merker 80.0 ist gleich Merker 80.1 (=Richtimpuls)

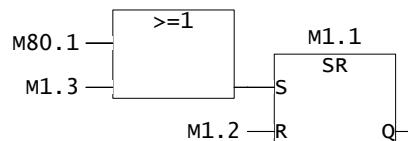
Setzen Merker 80.0

Beim 1. Durchlauf der AWL wird ein Impuls auf Merker 80.1 erzeugt.
Er kann verwendet werden um den Grundzustand zu erreichen



Netzwerk: 23 Beispiel für Verwendung des Richtimpuls

Merker M80.1 (=Richtimpuls) oder Merker 1.3 (=letzter Zustand, Schritt)
ist gleich Merker 1.1 (=Grundzustand, 1.Schritt)



Netzwerk: 24 Vergleichsoperationen Ganzzahlen (16 bit)

Vergleicht Merker MW12 mit Merker MW10

Die Merker MW10 und MW12 haben 16 bit Wortlänge

Mögliche Operationen:

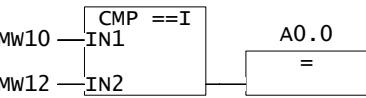
==I MW12 gleich MW10

<>I MW12 ungleich MW10

>I MW12 größer MW10
 <I MW12 kleiner MW10

>=I MW12 größer gleich MW10
 <=I MW12 kleiner gleich MW10

Wenn die Bedingung stimmt wird der Ausgang A0.0 auf 1 gesetzt

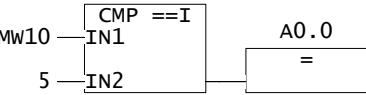


Netzwerk: 25 Vergleichsoperationen Ganzzahl (16 bit) mit Konstante

Vergleicht INT-Konstante mit Merker MW10

Der Merker MW10 hat 16 bit Wortlänge

Die INT-Konstante kann sich im Wert von -32768 bis +32768 bewegen
(INT = Integer)



Netzwerk: 26 Vergleichsoperationen Ganzzahlen (32 bit)

Vergleicht Merker MD12 mit Merker MD10

Die Merker MD10 und MD12 haben 32 bit Wortlänge

Mögliche Operationen:

==D MD12 gleich MD10

<>D MD12 ungleich MD10

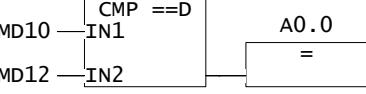
>D MD12 größer MD10

<D MD12 kleiner MD10

>=D MD12 größer gleich MD10

<=D MD12 kleiner gleich MD10

Wenn die Bedingung stimmt wird der Ausgang A0.0 auf 1 gesetzt

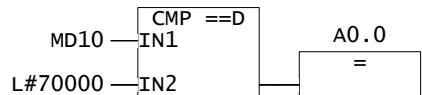


Netzwerk: 27 Vergleichsoperationen Ganzzahl (32 bit) mit Konstante

Vergleicht LINT-Konstante mit Merker MD10

Der Merker MD10 hat 32 bit Wortlänge

Die LINT-Konstante kann sich im Bereich von -2147483648 bis +2147483648 bewegen
 (LINT = Long Integer)



Netzwerk: 28 Vergleichsoperationen Gleitkommazahlen (16 bit)

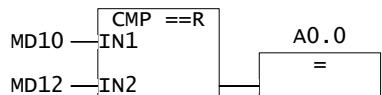
Vergleicht Merker MD12 mit Merker MD10
 Die Merker MD10 und MD12 sind 32bit-Gleitkommazahlen

Mögliche Operationen:
 ==I MD12 gleich MD10
 <>I MD12 ungleich MD10

>I MD12 größer MD10
 <I MD12 kleiner MD10

>=I MD12 größer gleich MD10
 <=I MD12 kleiner gleich MD10

Wenn die Bedingung stimmt wird der Ausgang A0.0 auf 1 gesetzt



Netzwerk: 29 Vergleichsoperationen Gleitpunktzahl (32 bit) und Konstante

Vergleicht Konstante mit Merker MD10
 Der Merker MD10 und die Konstante sind 32bit-Gleitkommazahlen

Die Gleitkomma-Konstante kann sich im negativen Bereich von $-1.175495e-38$ bis $-3.402823e+38$ bewegen und im positiven Bereich von $+1.175495e+38$ bis $+3.402823e+38$ bewegen.

