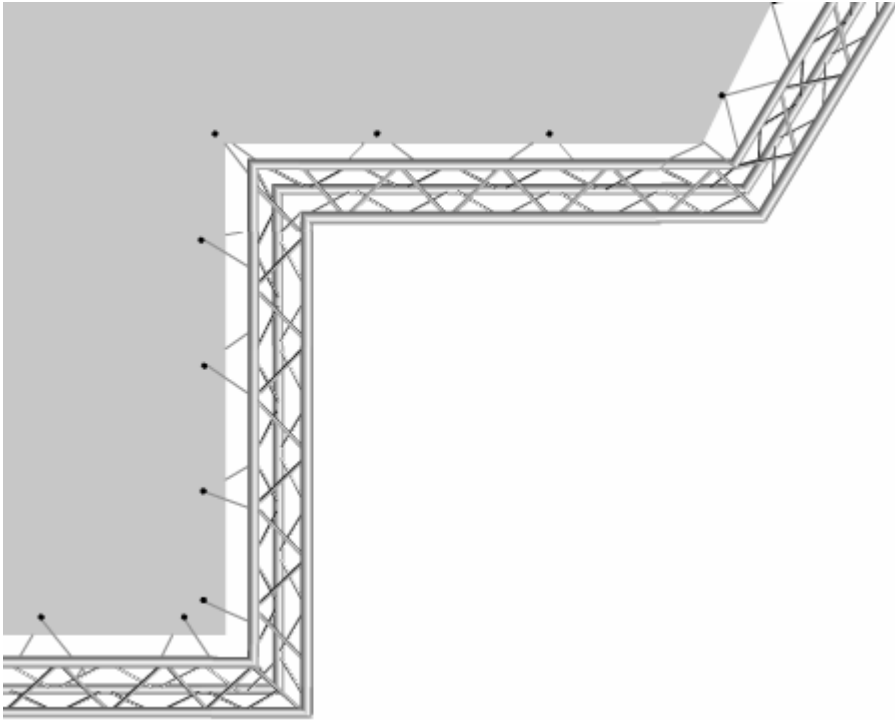
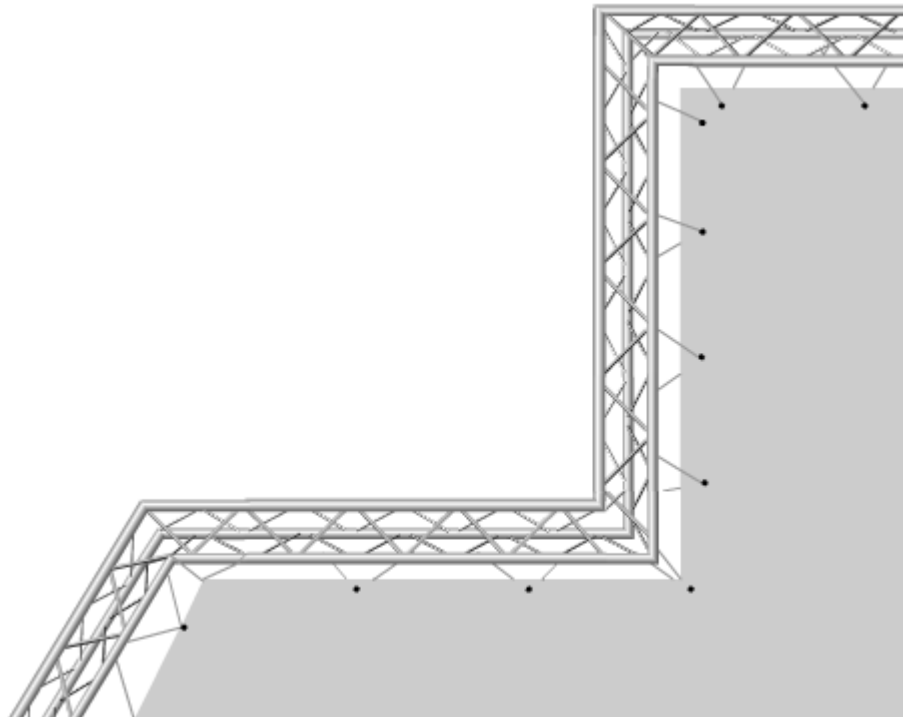




Programmable Logic Controllers



Inhoudsopgave

Inhoudsopgave.....	2
Inleiding.....	3
Opbouw van een sturing.....	3
Besturing met vaste bedrading.....	3
Besturen met PLC.....	3
Voordelen van PLC's.....	3
Opbouw en werking van de S7-212.....	4
Technische specificaties.....	4
De in- en uitgangen.....	5
Communicatie met de PC.....	5
Gestructureerd programmeren.....	6
Programmacyclus.....	6
Organisatiebouwsteen.....	6
Subroutines.....	6
Programmaweergaven.....	7
STL.....	7
Ladder.....	7
FBD.....	7
Programmavoorbeelden.....	7
Combinatorische logica.....	7
EN en NEN.....	7
OF en NOF.....	8
EXOF.....	8
Merkers.....	8
Geheugenfuncties.....	8
Flankdetectie.....	9
Sequentiële logica.....	9
Tijdsfuncties.....	9
Astabiele Multivibrator (versie 1).....	10
Astabiele Multivibrator (versie 2).....	10
Astabiele Multivibrator met veranderlijke duty cycle.....	11
Telfuncties.....	11
Vergelijkingsfuncties.....	11
Mixer van twee ingrediënten in een tank.....	12
Functiediagrammen.....	13
Principe.....	13
PLC programma opstellen.....	14
Voorbeeld: Oveninstallatie.....	16

Inleiding

Deze cursus is oorspronkelijk op- en samengesteld door ingenieur van der Linden V. en nadien bijgewerkt door Nick Electronics.

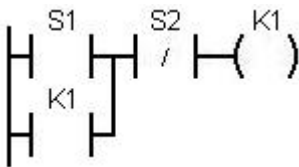
Opbouw van een sturing

Vroeger werden de besturingen voor automatische processen opgebouwd met relais of met elektronische schakelingen. 5NOF of NEN-poorten, flipflop's). Dit zijn zeer betrouwbare technieken maar ze hebben het nadeel niet flexibel te zijn (het is moeilijker om hardware aan te passen dan software), ontwikkeling en montage zijn zeer tijdrovend.

Door gebruik te maken van een PLC kan dit op een eenvoudige en snelle manier.

Besturing met vaste bedrading

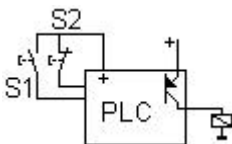
Nemen we als voorbeeld een start-stop schakeling.



De werking van de schakeling wordt bepaald door bedrading.

Besturen met PLC

Bij de PLC worden de bedieningsknoppen verbonden aan de ingangen. De spoel van de contactor wordt bediend via een uitgang van de PLC. De functionaliteit van de bedieningsknoppen wordt bepaald door de software in de PLC.



Voordelen van PLC's

- De Programmable Logic Controller of Programmeerbare Logische Sturing heeft als voordeel dat op een eenvoudige manier de schakeling kan aangepast worden door het programma te wijzigen. Zo kunnen ook de parameters van bijvoorbeeld tellers en timers veranderd worden.
- De bedrading is sterk gereduceerd en is eenvoudig. De montage is eenvoudig
- Men kan de schakeling testen door een simulatie, dit kan volledig buiten de installatie gebeuren zodat de productie geen hinder kan ondervinden. Storingzoeken wordt eenvoudiger door de ledindicatie van de in- en uitgangen.
- Door te werken met standaard PLC's wordt de kostprijs gereduceerd. Men kan naargelang de complexiteit van de installatie het aantal ingangen en uitgangen kiezen.

Opbouw en werking van de S7-212

Technische specificaties

Het type van PLC dat hier gebruikt wordt, wordt gevoed met een DC-spanning, de in- en uitgangen werken eveneens met een DC-spanning.

General Features

Physical size (L x W x D)	160 x 80 x 62 mm
Weight	0,3 kg
Power dissipation	5 W at 1,75 A load
User program size/storage	512words/EEPROM
User data size/storage	512words/RAM
Data retention	50 hr typical (8 hr minimum at 40° C)
Local I/O ¹	8 inputs / 6 outputs
Maximum number of expansion modules	2
Digital I/O supported	64 inputs / 64 outputs
Analog I/O supported	16 inputs / 16 outputs
Boolean execution speed	1,2 µs / instruction
Internal memory bits	128
Timers	64 timers
Counters	64 counters
High-speed counters	1 software (2 KHz max.)
Analog adjustments	1
Standards compliance	UL 508 CSA C22.2 142 FM Class I. Division 2 VDE 0160 compliant CE compliant



De in- en uitgangen

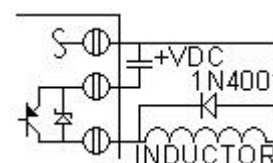
Input Points

Input type (IEC 1131-2)	Type 1 sinking
ON state range	15 – 30 VDC, 4 mA minimum
	35 VDC, 500 ms surge
ON state nominal	24 VDC, 7mA
OFF state maximum	5 VDC, 1mA
Response time I0.0 tot I0.7	0,3 ms maximum
Optical isolation	500 VAC, 1 min

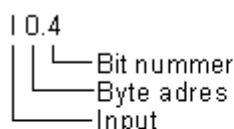
Output Points

Output type	Sourcing transistor
Voltage range	20,4 VDC tot 28,8 VDC
Maximum load current	0 to 40° C 55° C
Per single point	0,75 A 0,50 A
Per 2 adjacent points	1,00 A 0,75 A
All points total	2,25 A 1,75 A
Inductive load clamping	(per common)
Single pulse	2A L/R = 10ms
	1A L/R = 100ms
Repetitive	1W energy dissipation
	(1/2 Li² x switch rate __ 1W)
Leakage current	100µA
Switching delay	25 µs ON, 120 µs OFF
Surge current	4 A, 100ms
Voltage drop	1,8V maximum at maximum
current	
Optical isolation	500 VAC, 1 min
Short circuit protection	none

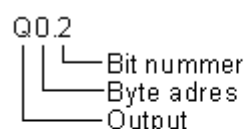
Wanneer men gebruikt maakt van gelijkspanning moeten de uitgangstrappen van de PLC beschermd worden tegen inductiespanningen van de relaispoelen. Dit kan met een diode (blusdiode) verholpen worden.



De aanduiding van de verschillende in- en uitgangen gebeurt met een byte en bit adressering.



1)	I0.0	I1.0
2)	I0.1	I1.1
3)	I0.2	I1.2
4)	I0.3	I1.3
5)	I0.4	I1.4
6)	I0.5	I1.5
7)	I0.6	I1.6
8)	I0.7	I1.7



1)	Q0.0	Q1.0
2)	Q0.1	Q1.1
3)	Q0.2	Q1.2
4)	Q0.3	Q1.3
5)	Q0.4	Q1.4
6)	Q0.5	Q1.5

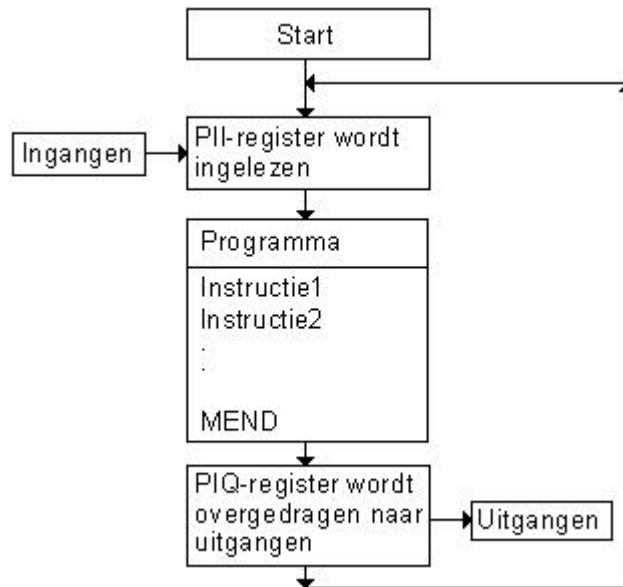
Communicatie met de PC

Het communiceren met de PLC gebeurt met een PC. Voor de seriële verbinding wordt een PC/PPI kabel gebruikt. Deze maakt de omvorming van RS-485 naar RS-232. De DIP-schakelaars worden op 9600 baud ingesteld.

Gestructureerd programmeren

Programmacyclus

De PLC gaat verschillende taken doen die cyclisch worden afgewerkt. Eén cyclus noemt men een scan cyclus. Deze cyclus bevat een aantal taken;



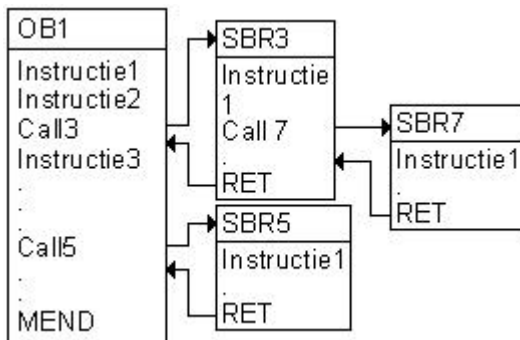
- Lezen van de ingangen
Elke cyclus begint met het lezen van de huidige toestand van de digitale ingangen en zet dit procesbeeld in het *process-image input register*.
- Uitvoering van het programma
De CPU start met de uitvoering van het programma. Hij begint met de eerste instructie en eindigt met de laatste. Wanneer in het programma een bepaalde uitgang verschillende keren wordt beïnvloed, gaat alleen de laatste beïnvloeding uitgevoerd worden.
- Schrijven naar de uitgangen
Op het einde van de scan cyclus gaat de CPU de waarden van het *processimage* – *output register* schrijven naar de digitale uitgangen.

Organisatiebouwsteen

Bij het opstarten van een PLC is alleen de organisatiebouwsteen OB1 actief. Als men uitsluitend deze bouwsteen gebruikt worden alle instructies cyclisch afgewerkt.

Subroutines

Bij het gestructureerd programmeren wordt er in modules geschreven. Deze modules (subroutines) worden opgeroepen vanuit OB1. Dit maakt dat het programma vlugger wordt afgewerkt en het testen vereenvoudigd wordt omdat het programma overzichtelijker wordt.



Programmaweergaven

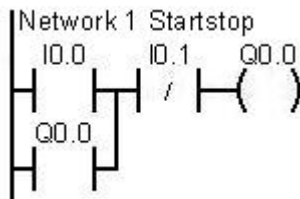
STL

Bij STL werkt men met mnemonische afkortingen

```
LD I0.0           //Als I0.0 ingeschakeld is
O Q0.0            //of Q0.0 aan is
AN I0.1           //en wanneer I0.1 niet ingeschakeld is
= Q0.0            //dan zal Q0.0 aan zijn
```

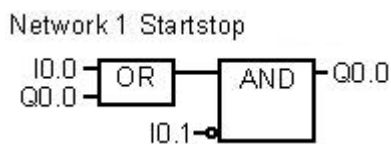
Ladder

Bij ladder wordt met contacten gewerkt.



FBD

Bij FBD wordt met logische functies gewerkt.

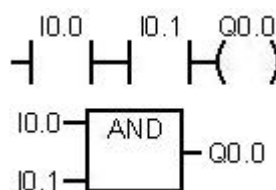


Programmavoorbeelden

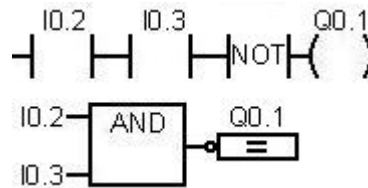
Combinatorische logica

EN en NEN

```
NETWORK1          //EN
LD I0.0
A I0.1
=Q0.0
```

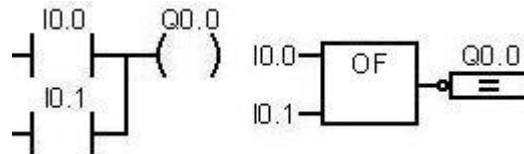


NETWORK2 //NEN
 LD IO.2
 A IO.3
 NOT
 = Q0.1

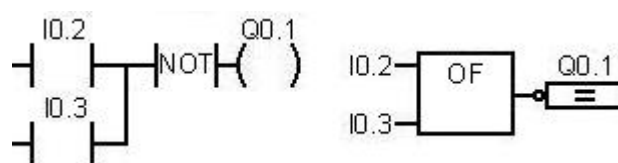


OF en NOF

NETWORK1 //OF
 LD IO.0
 O IO.1
 = Q0.0

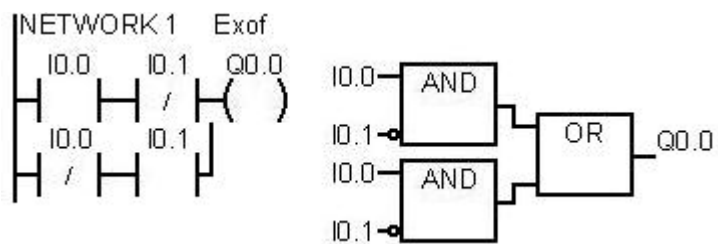


NETWORK2 //NOF
 LD IO.2
 O IO.3
 NOT
 = Q0.1



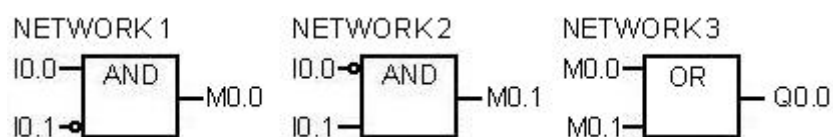
EXOF

NETWORK1 //EXOF
 LD IO.0
 AN IO.1
 LDN IO.0
 A IO.1
 OLD
 = Q0.0



Merkers

Soms is het nodig van tussenresultaten te bewaren. Daarvoor zijn er merkers beschikbaar. Merkers worden op dezelfde manier als in- of uitgangen geprogrammeerd. De merkers kunnen in andere delen van het programma verwerkt worden.



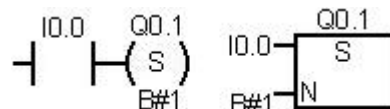
Geheugenfuncties

Om de toestand van een bepaalde actie te onthouden maakt men gebruik van een geheugenfunctie. De schakeling "onthoudt" dat er op de start of de stop is gedrukt bij een startstop-schakeling.

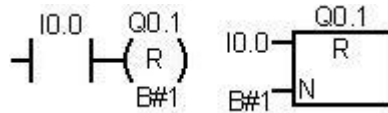
Zulke schakelingen kunnen met contacten of met poorten uitgevoerd worden.

Dit kan ook met een SET en RESET uitgevoerd worden.

NETWORK1 //SET
 LD IO.0
 S Q0.1,1

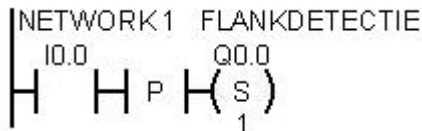


NETWORK2 //RESET
LD I0.1
R Q0.1,1



Flankdetectie

Door bij het symbool van een voorwaarde letter P of N te plaatsen gaat er een puls van één cyclus gegenereerd worden, respectievelijk bij een stijgende of dalende flank. Bij STL maakt men gebruik van de EU instructie (Edge Up).



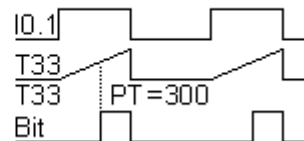
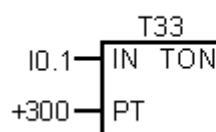
Sequentiële logica

Tijdsfuncties

Men heeft twee tijdsfuncties; de TON en de TONR. Beide tijdsfuncties tellen pulsen op als de toestand van de ingang hoog is, ze tellen niet meer verder als de ingang laag is. Bij de TON-timer wordt de timer gereset als de ingang laag wordt, bij de TONR-timer stopt het tellen maar er wordt niet gereset, als de ingang terug hoog wordt telt hij verder. Wanneer de TONR-timer moet gereset worden dan moet dit met een afzonderlijke resetinstructie.

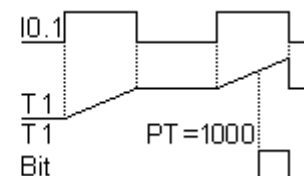
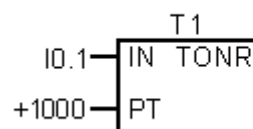
NETWORK1
LD I0.1
TON T33, 300

NETWORK1 Ton-Timer



NETWORK1
LD I0.1
TONR T1, 1000

NETWORK1 Ton-Timer



De PT staat voor Preset Time. Dit is het aantal te tellen pulsen. De periode of de resolutie wordt bepaald door het Timernummer.

Timer	Nummer	Resolutie	Max. waarde
TON	T32	1ms	32,767s
	T33 tot T36	10ms	327,67s
	T37 tot T63	100ms	3276,7s
TONR	T0	1ms	32,767s
	T1 tot T4	10ms	327,67s
	T5 tot T31	100ms	3276,7s

Inschakelvertraging:

NETWORK1 //Inschakelvertraging

LD I0.1
TON T33, +300

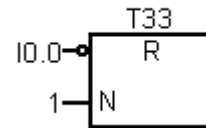
NETWORK2

LDN I0.0
R T33, 1

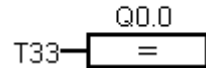
NETWORK3

LD T33
= Q0.0

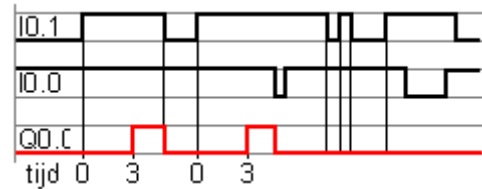
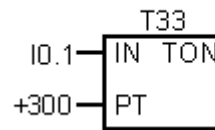
NETWORK 2



NETWORK 3



NETWORK 1 inschakelvertraging

Uitschakelvertraging:

NETWORK1 //Uitschakelvertraging

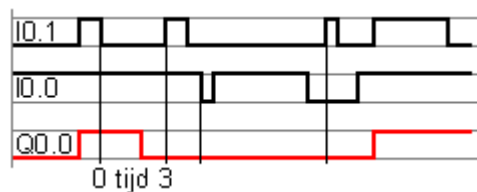
LD I0.1
EU
S Q0.0, 1

NETWORK2

LDN I0.1
TON T33, +300

NETWORK3

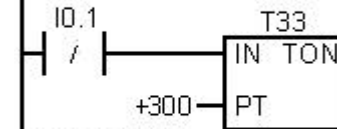
LD T33
ON I0.0
R Q0.0, 1



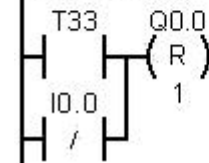
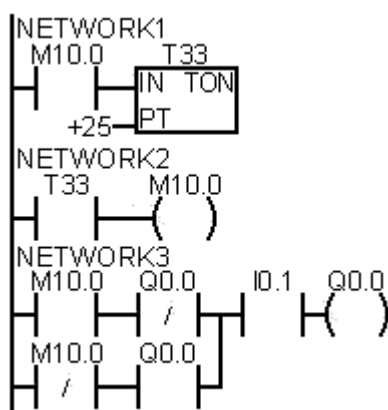
NETWORK 1



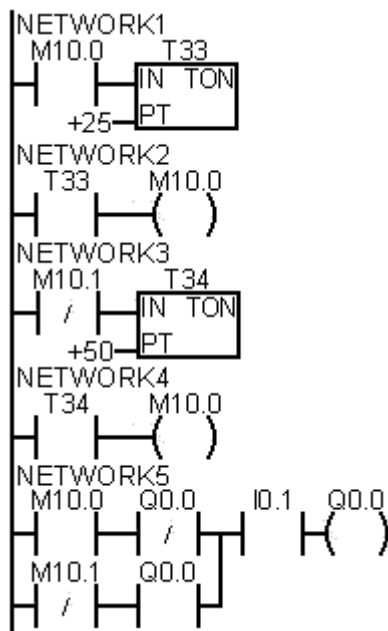
NETWORK 2



NETWORK 3

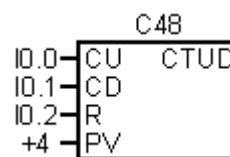
**Astabile Multivibrator (versie 1)****Astabile Multivibrator (versie 2)**

Astabile Multivibrator met veranderlijke duty cycle



Telfuncties

C48: nummer teller
 CU: Count Up (Tel Op)
 CD: Count Down (Tel Af)
 R: Reset
 CTUD: CounTer Up Down



Vergelijkingsfuncties

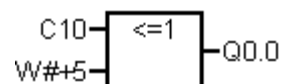
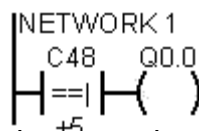
Men kan bepaalde grootheden (tellerwaarden, tijdswaarden...) gaan vergelijken met vergelijkingsfuncties.

X == I Y Als X gelijk is aan Y is de voorwaarde waar.
 X >= I Y Als X groter is dan of gelijk aan Y is de voorwaarde waar.
 X <= I Y Als X kleiner is dan of gelijk aan Y is de voorwaarde waar.

De letter I staat voor Integer getal. Men kan ook gaan vergelijken met een byte (B), een woord (W), een dubbel woord (D) of een reëel getal (R).

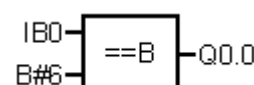
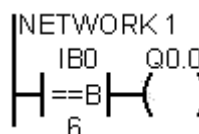
De uitgang wordt actief als de teller gelijk is aan 5.

LDW <= C10, W#+5
 = Q0.0



Men kan een bepaalde waarde ingeven via de binaire ingangen.
 Wanneer de waarde 6 wordt ingesteld is de voorwaarde voldaan.

LDB = IB0, B#6
 = Q0.0

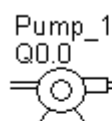
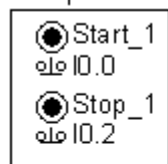


Mixen van twee ingrediënten in een tank

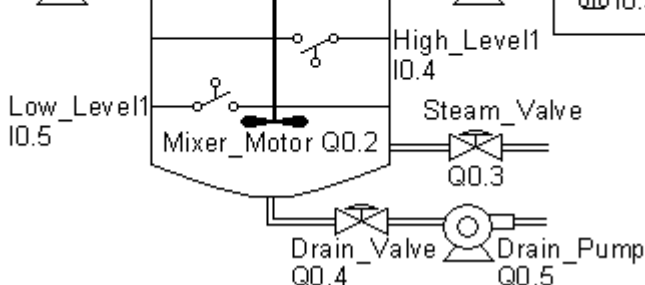
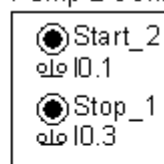
In deze toepassing maakt men gebruik van symbolische adressen. Deze tabel kan men invoeren via de "Symbol Table".

Start_1	I0.0	Start voor ingrediënt 1
Start_2	I0.1	Start voor ingrediënt 2
Stop_1	I0.2	Stop voor ingrediënt 1
Stop_2	I0.3	Stop voor ingredient 2
Hoog_niveau	I0.4	Contact voor hoog niveau
Laag_niveau	I0.5	Contact voor laag niveau
Reset	I0.7	Reset controle voor teller
Pomp_1	Q0.0	Pomp voor ingrediënt 1
Pomp_2	Q0.1	Pomp voor ingredient 2
Mixer_motor	Q0.2	Motor tank mixer
Stoom_klep	Q0.3	Stoom voor de verwarming van de mix
Uitlaat_klep	Q0.4	Uitlaat van de tank
Uitlaat_pomp	Q0.5	pomp voor het ledigen tank
Ho_niv_bereikt	M0.1	Geheugen bit
Mix_timer	T37	Voor controle mixen/ opwarmen mix
Cyclus_teller	C30	Tellen van aantal cycli

Pump 1 Controls

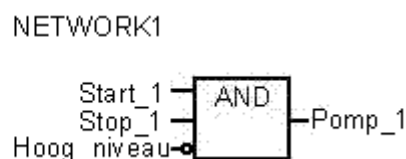
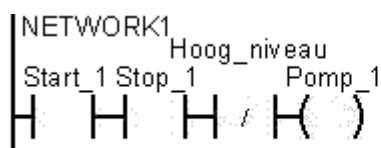


Pump 2 Controls



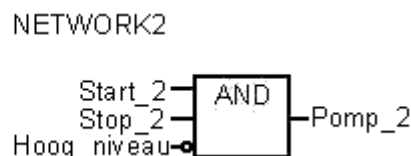
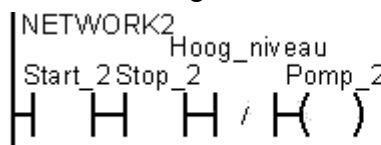
NETWORK1
LD Start_1
A Stop_1
AN Hoog_niveau
= Pump_1

//Vul de tank met ingrediënt 1



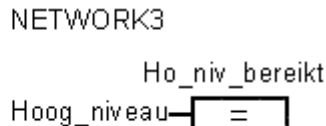
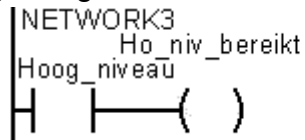
NETWORK2
LD Start_2
A Stop_2
AN Hoog_niveau
= Pump_2

//Vul de tank met ingrediënt 2



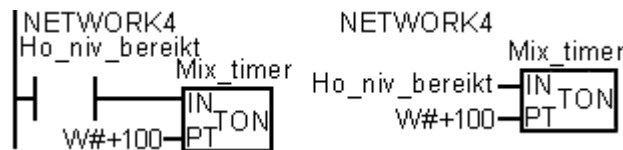
NETWORK3
LD Hoog_niveau
= Ho_niv_bereikt

//Zet geheugenbit als het hoge niveau bereikt is



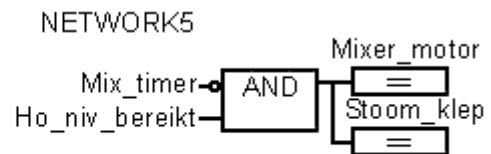
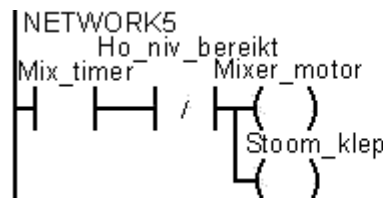
NETWORK4 //Start de timer als het hoge niveau bereikt is

LD Hoog_niveau
TON Mix_timer, +100



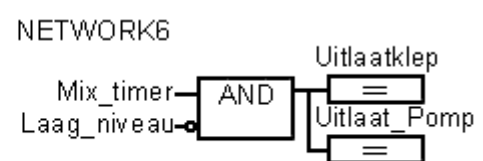
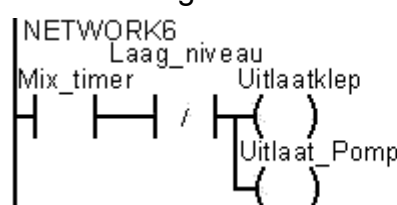
NETWORK5 //Zet de mix motor aan

LDN Mix_timer
A Ho_niv_bereikt
= Mixer_motor
= Stoom_klep



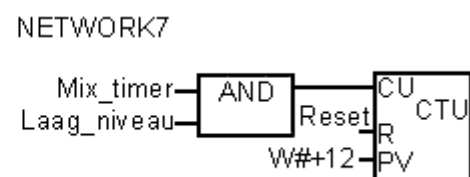
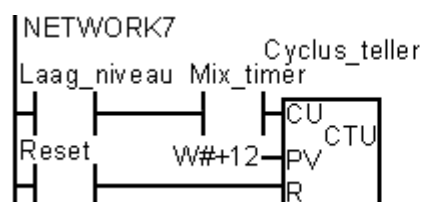
NETWORK6 //Maak de tank leeg

LD Mix_timer
AN Laag_niveau
= Uitlaat_klep
= Uitlaat_pomp



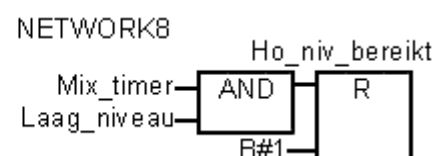
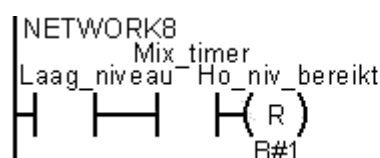
NETWORK7 //Tel elke cyclus

LD Laag_niveau
A Mix_timer
LD Reset
CTU Cyclus_teller, +12



NETWORK8 //Reset de geheugenbit bij lag niveau en einde timer

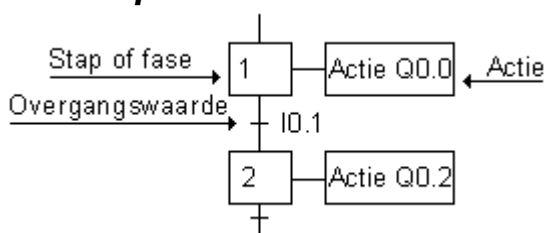
LD Laag_niveau
A Mix_timer
R Ho_niv_bereikt, 1



Functiediagrammen

Men tracht met een grafische voorstelling een duidelijk leesbare structuur te verkrijgen die door zoveel mogelijk mensen kan begrepen worden. Het functiediagram werkt onafhankelijk van de gebruikte sturingstechniek; elektrische schakelingen, digitale, pneumatische of PLC technieken.

Principe



Men maakt gebruik van de volgende basisregels:

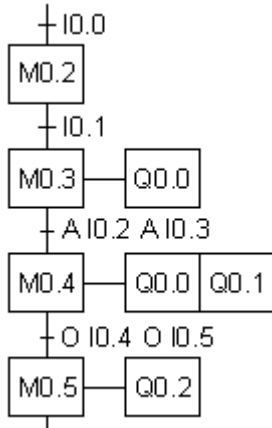
- Het proces wordt in fasen onderverdeeld. Deze fasen worden één na één actief.
- Aan iedere fase kunnen één of meerdere acties gekoppeld worden.

- Tussen de fasen is er een overgang. Dit gaat gepaard met een overgangswaarde welke de overgang vergrendelt zolang niet aan deze voorwaarde voldaan is.
- De volgende fase wordt actief als de vorige fase actief is **én** er aan de overgangsvoorwaarden voldaan wordt.

PLC programma opstellen

Elke fase uit het functiediagram wordt aan een merker toegewezen.

Een volgende fase wordt actief als de vorige fase actief is en er aan de overgangsvoorwaarden voldaan is. Het gevolg van het actief worden van deze fase is dat de vorige fase niet-actief wordt.



NETWORK3

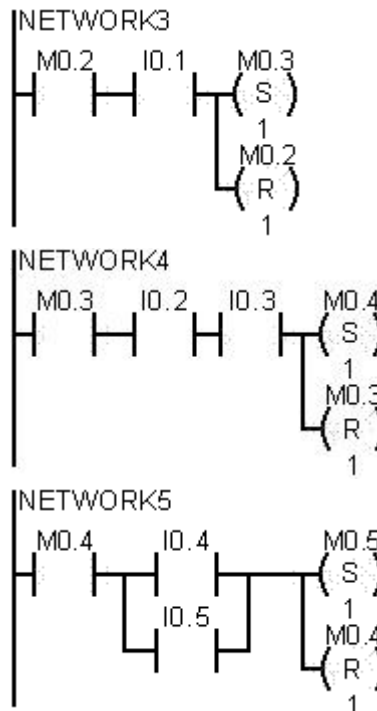
```
LD M0.2
A IO.1
S M0.3, 1
R M0.2, 1
```

NETWORK4

```
LD M0.3
A IO.2
A IO.3
S M0.4, 1
R M0.3, 1
```

NETWORK5

```
LD M0.4
LD IO.4
O IO.5
ALD
S M0.5, 1
R M0.4, 1
```



Opmerking: initialisatie

Bij het opstarten van de sturing is de initiële fase actief. Dit is de rusttoestand van de installatie. Alle andere fasen zijn gedurende de eerste cyclus niet actief.

Wanneer men een PLC-programma opstelt moet men er voor zorgen dat deze begintoestand geactiveerd wordt.

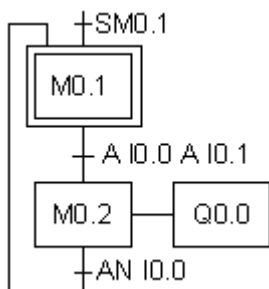
Dit gebeurt bij de eerste cyclus. Vele toestellen zijn voorzien van een extra geheugenbit die gedurende de eerste cyclus van de PLC actief is. Indien deze merker niet aanwezig is dan moet men deze zelf programmeren.

Bij de S7-200 serie heeft men een aantal speciale geheugenbit waaronder SM0.1. Deze geheugenbit is actief bij de eerste programmacycclus

SM0.0	Altijd aan.
SM0.1	Eerste cyclus aan.
SM0.3	Power up
SM0.4	30 sec OFF / 30 sec ON
SM0.5	0,5 sec OFF / 0,5 sec ON
SM0.6	1 scan OFF / 1 scan ON

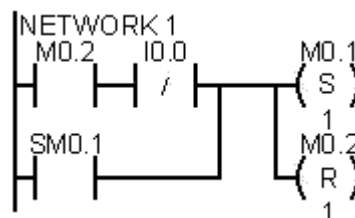
Een motor wordt geactiveerd als de start wordt ingedrukt, als men op de stop drukt dan valt de motor stil.

I0.0: stop I0.1: start Q0.0: motor



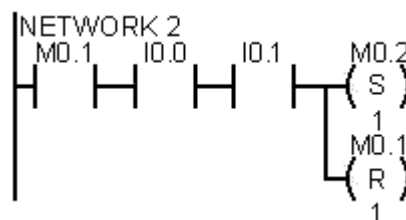
NETWORK1

```
LD M0.2
AN I0.0
O SM0.1
S M0.1, 1
R M0.2, 1
```



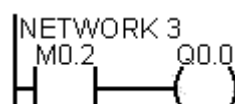
NETWORK2

```
LD M0.1
A I0.0
A I0.1
S M0.2, 1
R M0.1, 1
```



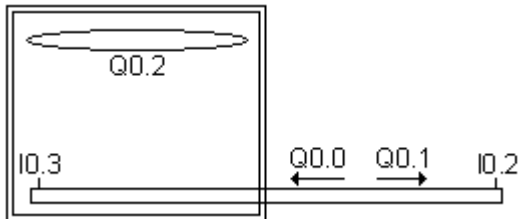
NETWORK3

```
LD M0.2
= Q0.0
```

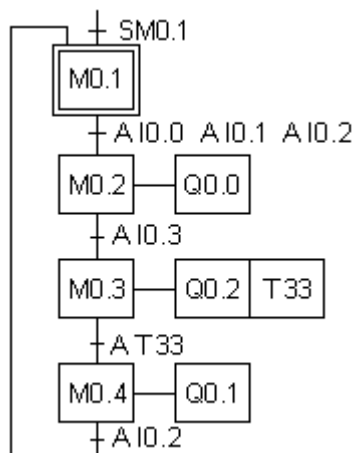


Voorbeeld: Oveninstallatie

We beschouwen een oveninstallatie met aan en afvoer door een transportband die in twee zinnen kan bewegen.

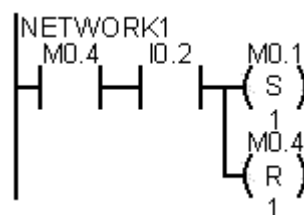


IO.0	Stopdrukknop met normaal gesloten contact
IO.1	Startdrukknop
IO.2	Sensor voor stukken rechts op band
IO.3	Sensor voor stukken links op band in de oven
Q0.0	Band draait links
Q0.1	Band draait rechts
Q0.2	Verwarming



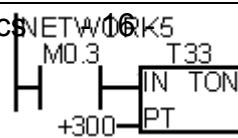
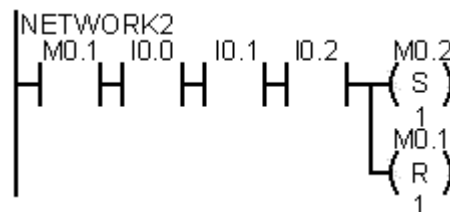
NETWORK1

```
LD M0.4
A IO.2
S M0.1, 1
R M0.4, 1
```



NETWORK2

```
LD M0.1
A IO.0
A IO.1
A IO.2
S M0.2, 1
R M0.1, 1
```



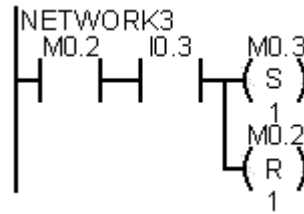
NETWORK3

LD M0.2

A I0.3

S M0.3, 1

R M0.2, 1



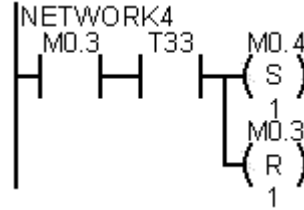
NETWORK4

LD M0.3

A T33

S M0.4, 1

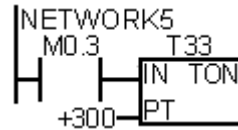
R M0.3, 1



NETWORK5

LD M0.3

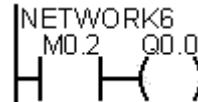
TON T33, +300



NETWORK6

LD M0.2

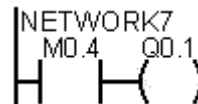
=Q0.0



NETWORK7

LD M0.4

= Q0.1



NETWORK8

LD M0.3

= Q0.2



NETWORK9

LD SM0.1

ON I0.0

S M0.1, 1

