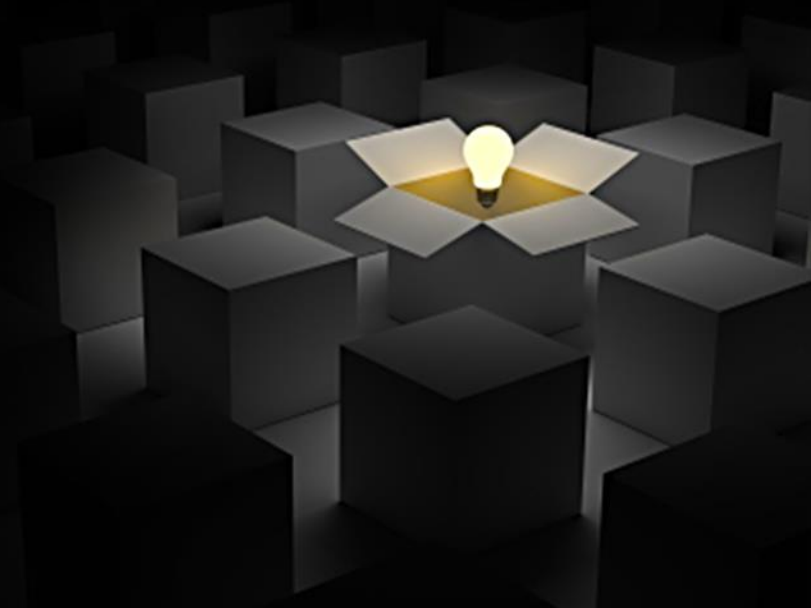


Why PLC?

Voe wadde?

Waarom?



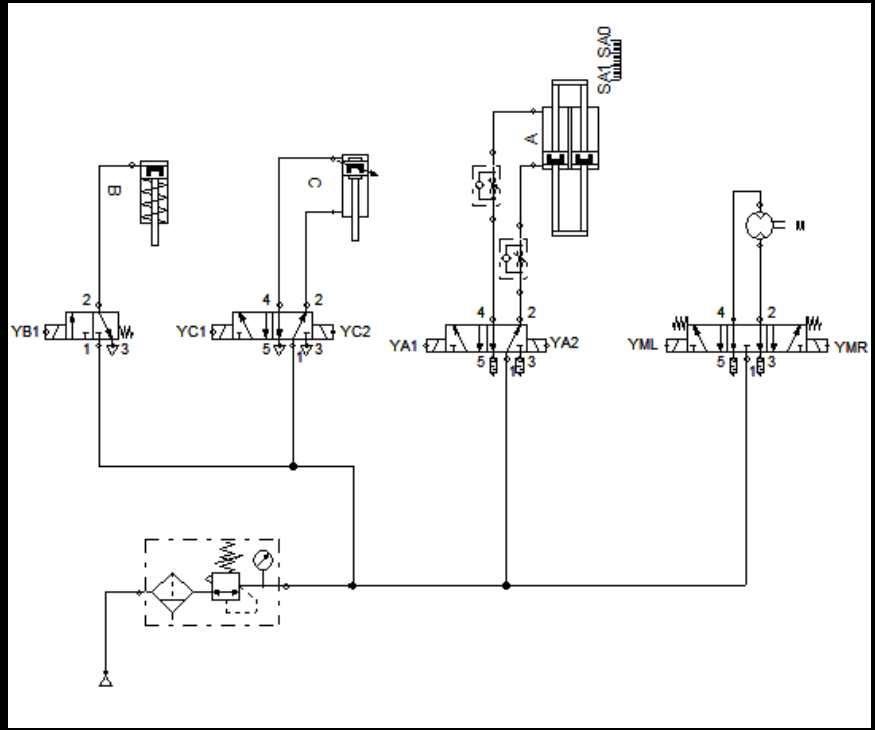
Probleem!

- Elektropneumatica voorbeeld

Voorbeeld uitgewerkt in vorige ppt

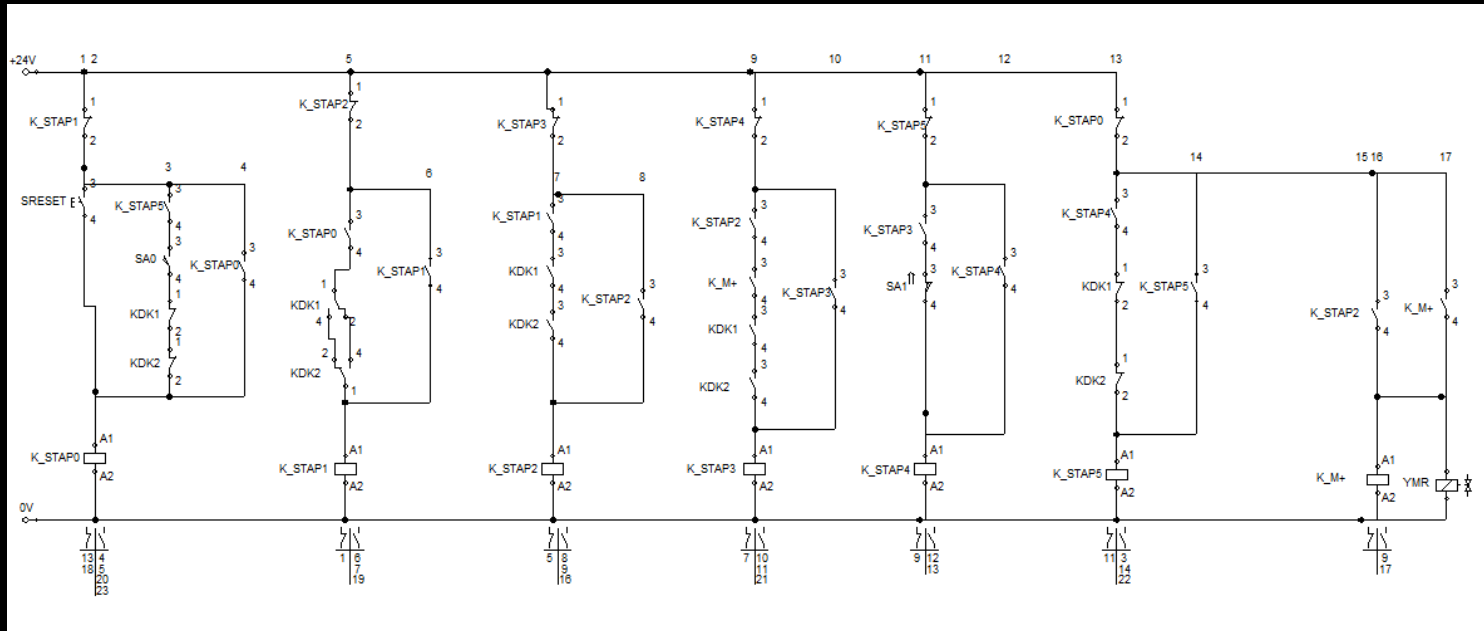


Problem!



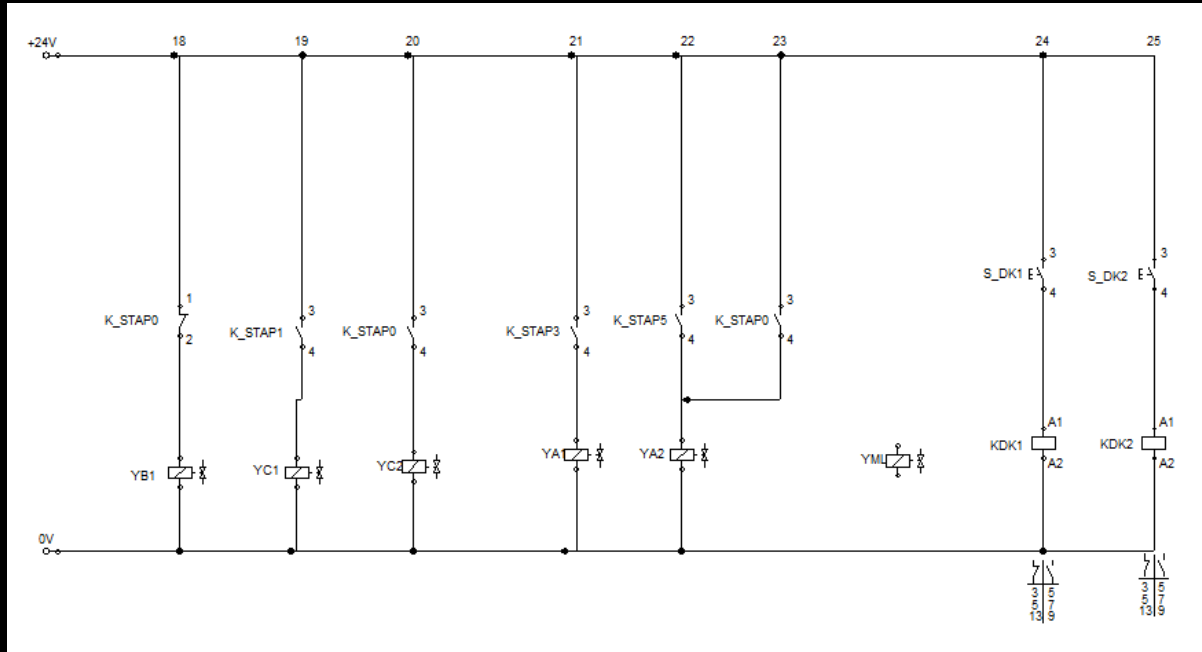
Problem!

- Elektropneumatica: oplosmethode 1 – deel 1



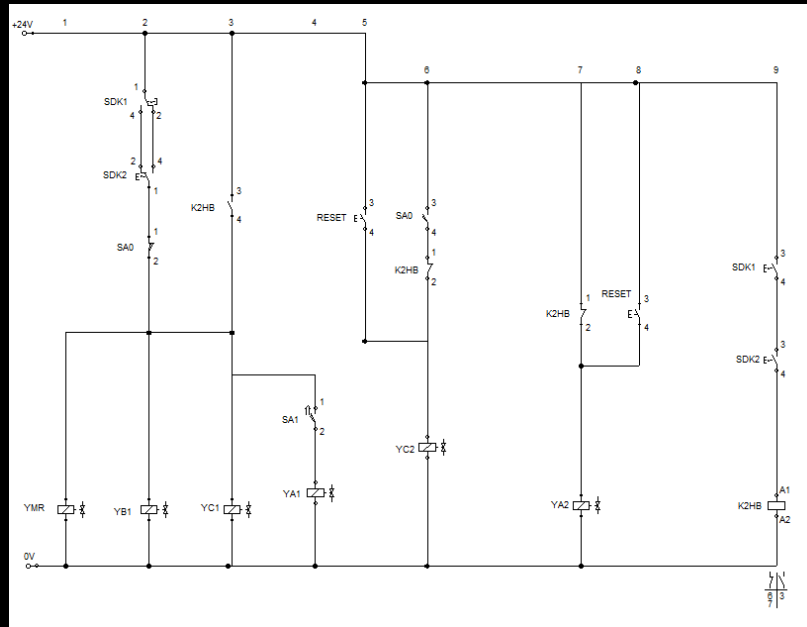
Problem!

- Elektropneumatica: oplosmethode 1 – deel 2



Problem!

- Elektropneumatica: oplosmethode 2

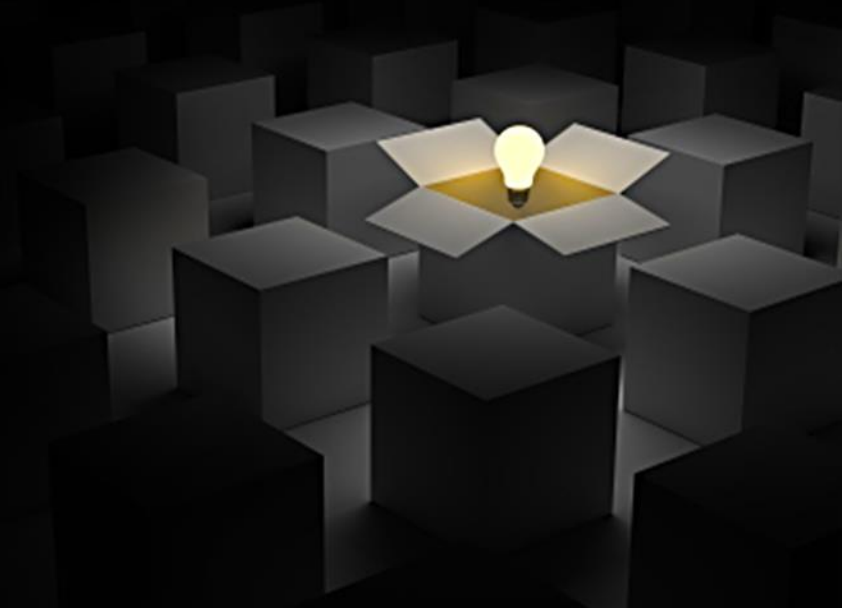
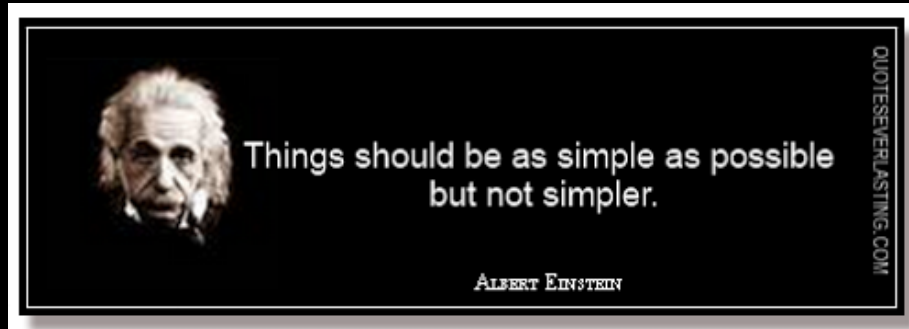


Probleem!

- Elektropneumatica @sequentiële sturing
 - Te complexe schakelingen
 - Te veel
 - Hulprelais
 - hulpcontacten voor eindeloop en/of schakelaars
 - Weinig flexibel
 - Veel kableerwerk (duur qua manuren)



Idea? Idee? Idée?



Oplossing? → Oplossing!

- Introductie 'slim' apparaat
 - Schakeling 'programmeren'
 - *Flexibel (& wijzigbaar)*
 - *Reductie (fysieke) hulpcontacten*
 - *Minder kableerwerk*
 - *Wél: meer programmeertijd nodig*



Welk 'slim' apparaat???

P L C

Programmable

Logic

Controller



Wat met de 'rest' uit de schakeling?

NODIG

- Schakelaars en drukknoppen
- Eindelopen en benaderingsensoren
- Ventielen (cilinders)
- LED's e.a. indicatoren
- Contactoren (motoren, drives)
- (hulpcontacten kunnen)

Reductie

- ~~Hulp~~relais
- ~~Meerdere~~schakelcontacten

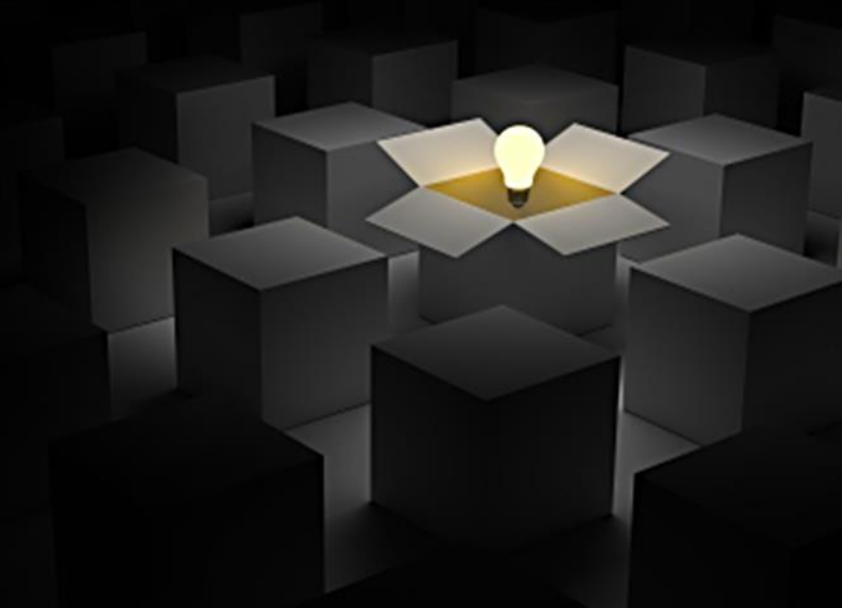


Which PLC?

Wufferen?

Witch?

Welke PLC?



Welke PLC?

Hoe kies je een PLC?

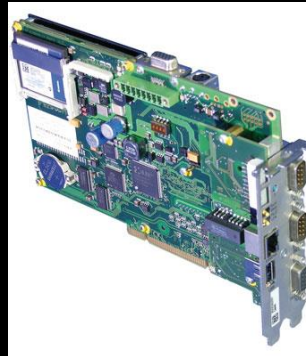
- Inbouwmethode
- Configuratie



Welke PLC?

Inbouwmethode

- Traditionele PLC
 - Compacte PLC
 - Rack PLC
 - Modulair = uitbreidbaar
- Touch PLC
- *Safety PLC*
- Slot PLC
- Soft PLC



Welke PLC?

Hoe kies je een PLC?

Configuratie

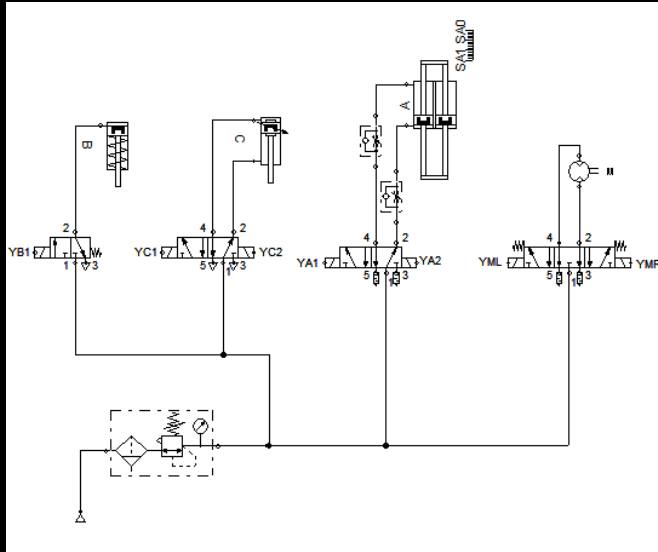
- Snelheid verwerking
- Grootte geheugen
- Communicatiepoorten
- # DI
- # DO
- # AI
- # AO
- Speciale IO?



Welke PLC?

Hoe kies je een PLC?

- Bij voorbeeld?



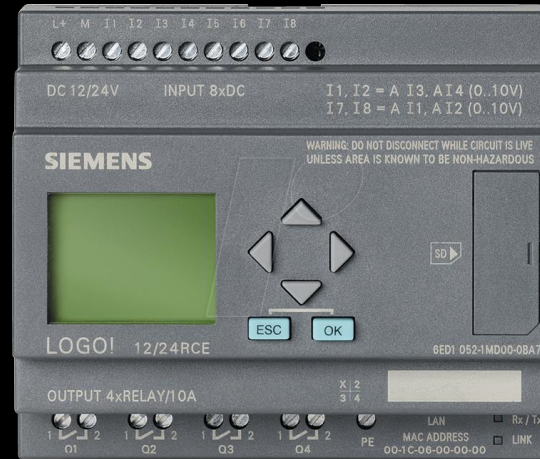
- Snelheid verwerking:
 - NVT: niet van toepassing
- Grootte geheugen:
 - NVT
- Communicatiepoorten:
 - NVT – stand alone
- # DI = 5
 - 2 eindelooptschakelaars
 - Reset
 - (2handenbediening zou kunnen)
=> beter inlezen 2HB-relais
- # DO = 7
 - 4 ventielen: 7 gestuurde zijden.
- # AI
 - NVT
- # AO
 - NVT
- Speciale IO?
 - NVT



Welke PLC?

Hoe kies je een PLC?

- Bij voorbeeld: Siemens Logo?
 - Spanning CPU!!!
 - # DI:
 - 8 on board
 - 8 > 5
 - OK
 - # DO:
 - 4 on board
 - 4 < 7
 - Onvoldoende
 - Extra outputkaart? + 4: OK



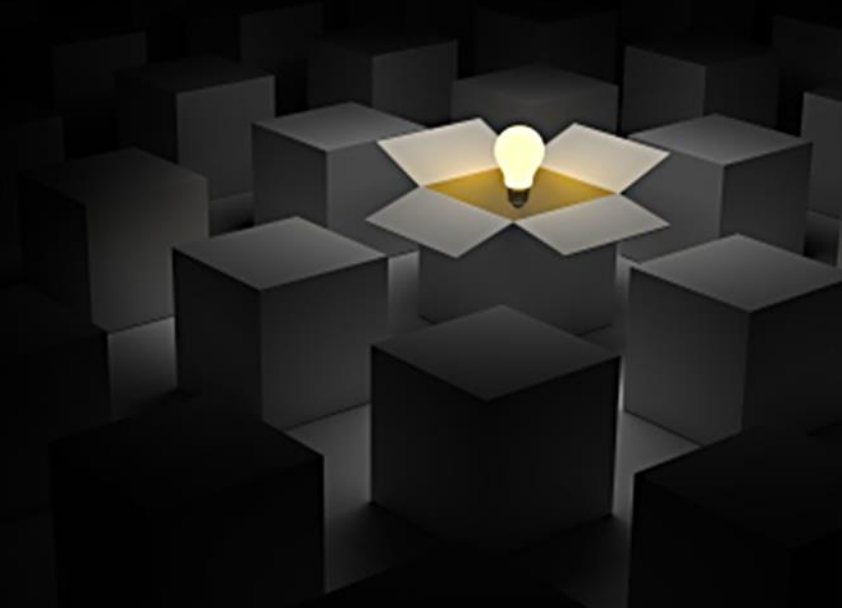
Welke PLC?

Hoe kies je een PLC?

- Bij voorbeeld: S7-1214?
 - Spanning CPU!!!
 - # DI:
 - 14 on board
 - $14 > 5$
 - OK
 - # DO:
 - 10 on board
 - $10 > 7$
 - OK

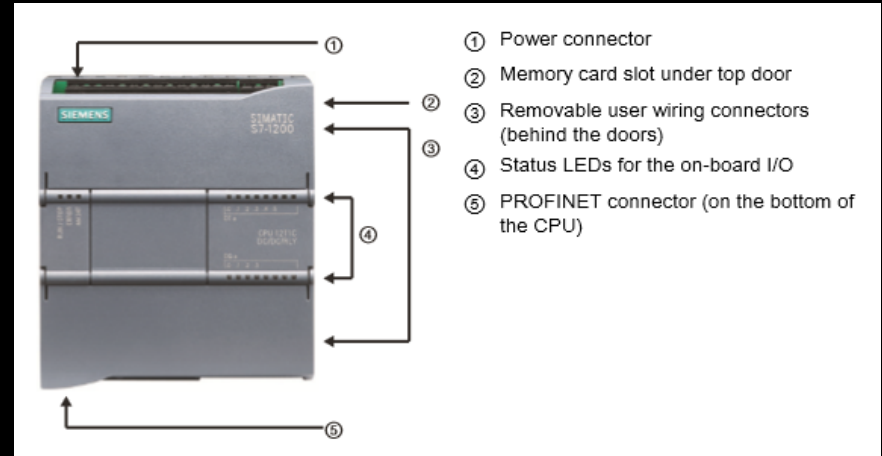


Type PLC's @VTI Brugge?



S7-1214C AC/DC/RLY

- Aanwezig in B28 (12-tal)



S7-1214C AC/DC/RLY



Feature		CPU 1214C
Physical size (mm)		110 x 100 x 75
User memory	Work	75 Kbytes
	Load	4 Mbytes
	Retentive	10 Kbytes
Local on-board I/O	Digital	14 inputs/10 outputs
	Analog	2 inputs
Process image size	Inputs (I)	1024 bytes
	Outputs (Q)	1024 bytes
Bit memory (M)		8192 bytes
Signal module (SM) expansion		8
Signal board (SB), Battery board (BB), or communication board (CB)		1
Communication module (CM) (left-side expansion)		3
High-speed counters	Total	6
	Single phase	3 at 100 kHz
		3 at 30 kHz
Quadrature phase	3 at 80 kHz	
	3 at 20 kHz	
Pulse outputs ¹		4
Memory card		SIMATIC Memory card (optional)
Real time clock retention time		20 days, typ. / 12 day min. at 40 degrees C (maintenance-free)
PROFINET		1 Ethernet communication port
Real math execution speed		2.3 μs/instruction
Boolean execution speed		0.08 μs/instruction



S7-1214C AC/DC/RLY

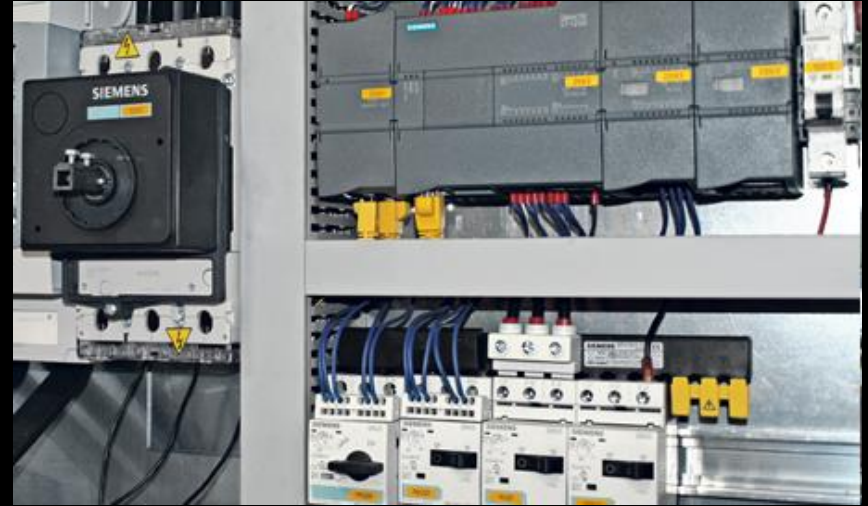
Table 1-2 Blocks, timers and counters supported by S7-1200

Element	Description	
Blocks	Type	OB, FB, FC, DB
	Size	30 Kbytes (CPU 1211C) 50 Kbytes (CPU 1212C) 64 Kbytes (CPU 1214C and CPU 1215C)
	Quantity	Up to 1024 blocks total (OBs + FBs + FCs + DBs)
	Address range for FBs, FCs, and DBs	1 to 65535 (such as FB 1 to FB 65535)
	Nesting depth	16 from the program cycle or start up OB; 4 from the time delay interrupt, time-of-day interrupt, cyclic interrupt, hardware interrupt, time error interrupt, or diagnostic error interrupt OB
	Monitoring	Status of 2 code blocks can be monitored simultaneously
	OBs	Program cycle
Startup		Multiple: OB 100, OB 200 to OB 65535
Time-delay interrupts and cyclic interrupts		4 ¹ (1 per event): OB 200 to OB 65535
Hardware interrupts (edges and HSC)		50 (1 per event): OB 200 to OB 65535
Time error interrupts		1: OB 80
Diagnostic error interrupts		1: OB 82
Timers	Type	IEC
	Quantity	Limited only by memory size
	Storage	Structure in DB, 16 bytes per timer
Counters	Type	IEC
	Quantity	Limited only by memory size
	Storage	Structure in DB, size dependent upon count type <ul style="list-style-type: none">• SInt, USInt: 3 bytes• Int, UInt: 6 bytes• DInt, UDInt: 12 bytes

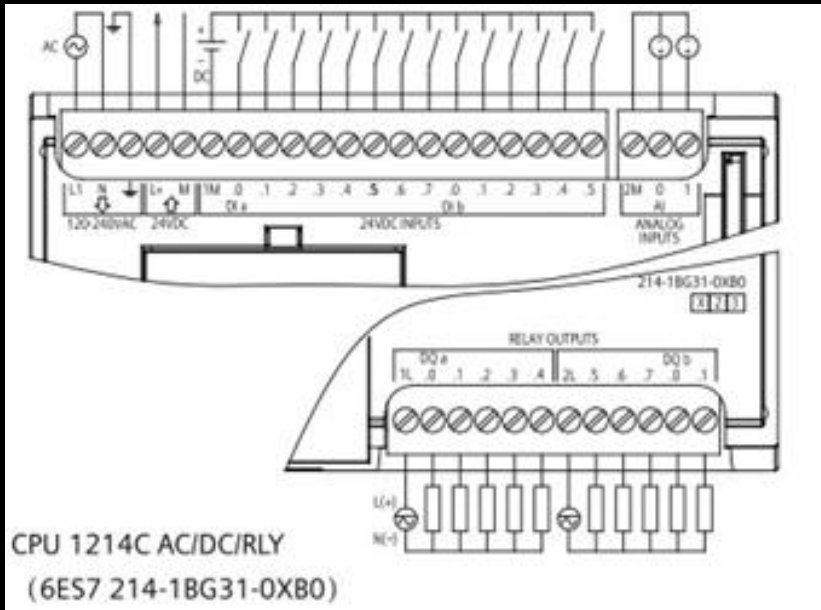
¹ Time-delay and cyclic interrupts use the same resources in the CPU. You can have only a total of 4 of these interrupts (time-delay plus cyclic interrupts). You cannot have 4 time-delay interrupts and 4 cyclic interrupts.



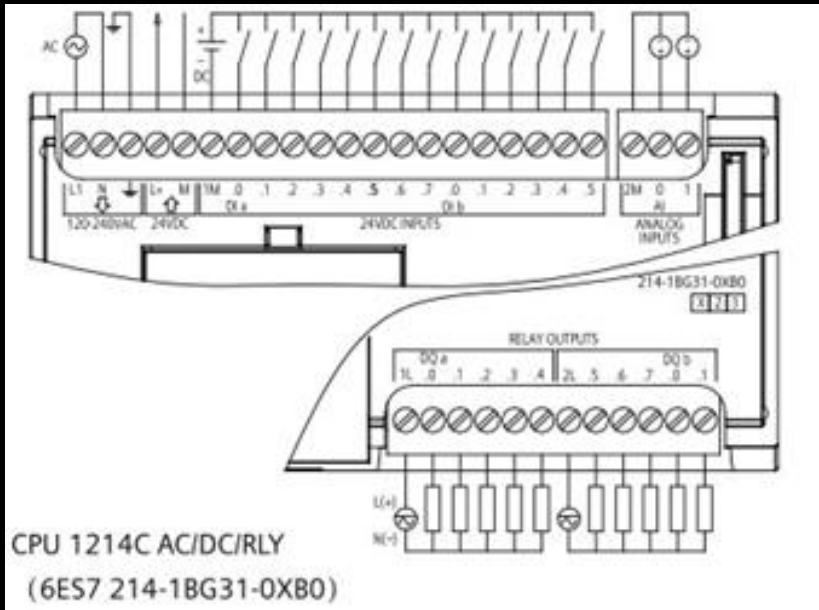
S7-1214C ac/dc/rly



S7-1214C ac/dc/rly



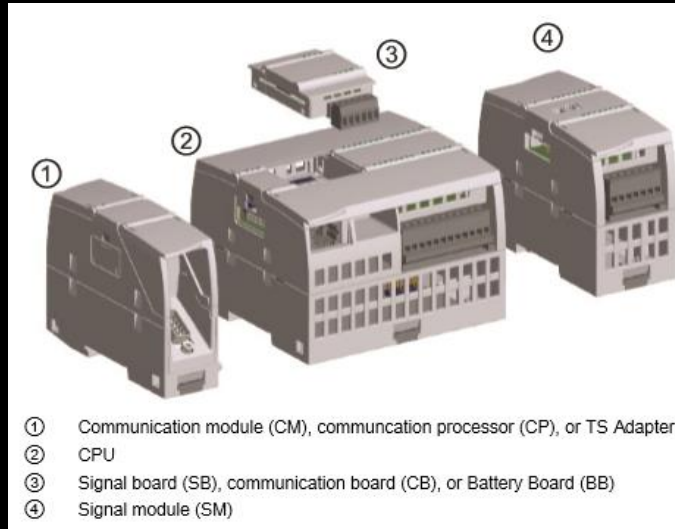
S7-1214C ac/dc/rly



- **Communicatie**
 - Profinet
 - Industrial ethernet
- **Voeding:** 230V ac
- aarding
- **Bron** 24V dc
 - Zwak 0,2A
 - Niet gebruiken bij aanwezigheid andere bron
- **14 DI** (sinking, 24Vdc)
- **2 AI**
- **10 DO** (sourcing, RLY)



Handleiding S7-1200 – Easy Book



- ① Communication module (CM), communication processor (CP), or TS Adapter
- ② CPU
- ③ Signal board (SB), communication board (CB), or Battery Board (BB)
- ④ Signal module (SM)



Mogelijkheid toevoegen signaalbord!



Soorten Digitale ingangen? ALGEMEEN

- DI => intern mechanisme PLC.
- Kan niet gewijzigd worden.
- Belang kennis soorten DI?
 - Aansluitmethode sensoren



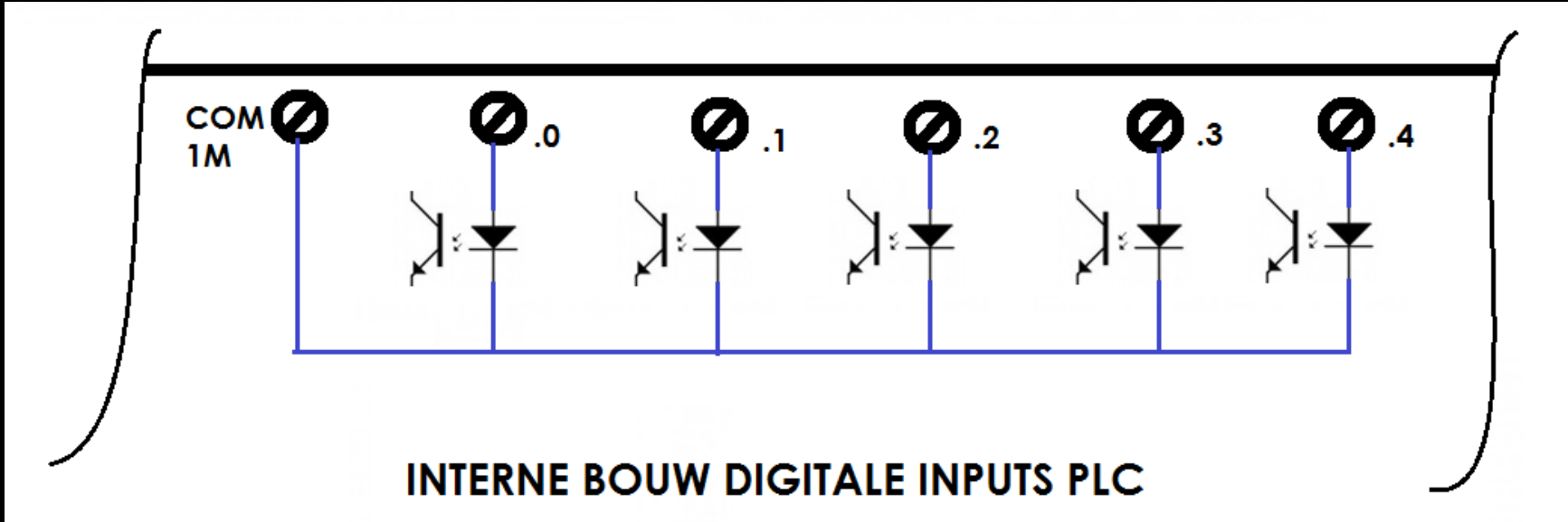
Soorten Digitale ingangen? ALGEMEEN

- Digitale ingang = opto-coupler
 - Reden? → Galvanische scheiding

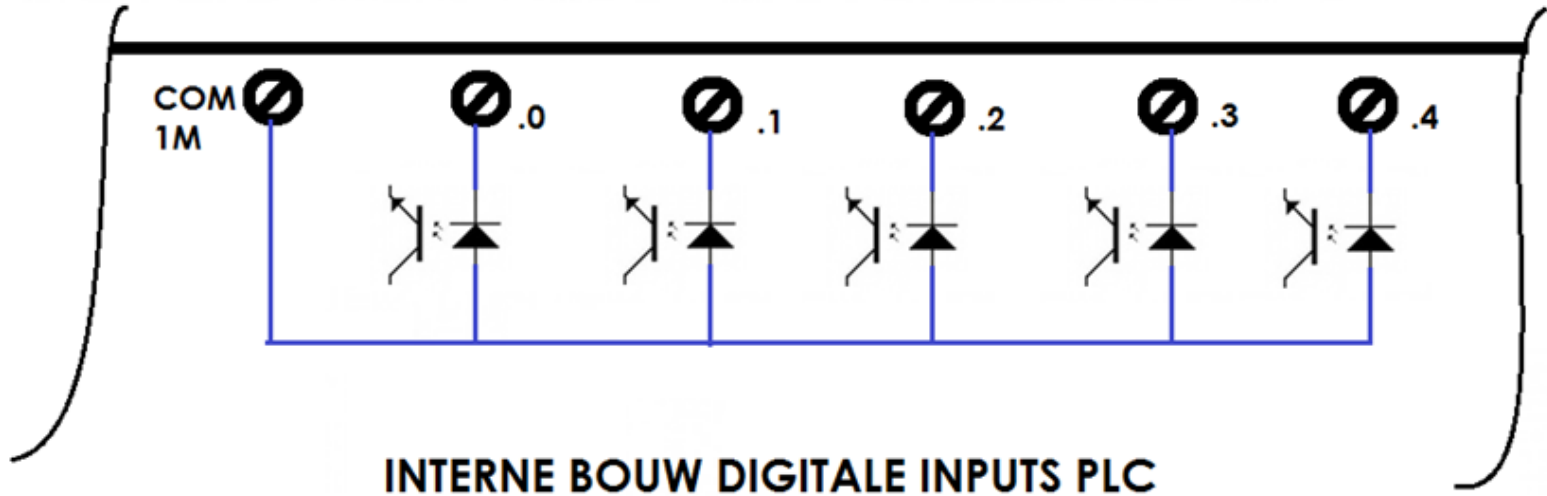
- 
- Sinking DI? → Stroom vloeit **IN** de DI
 - Sourcing DI? → Stroom vloeit **UIT** de DI



Digitale ingangen: Sinking

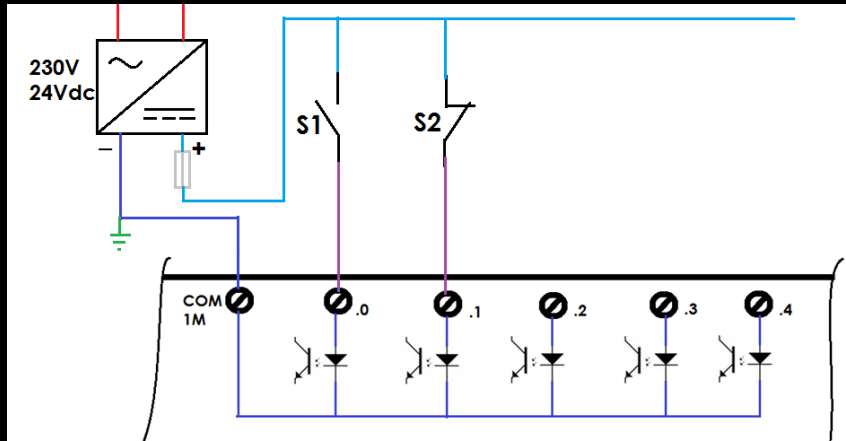


Digitale ingangen: Sourcing



Digitale ingangen: schakelaar

DI: sinking



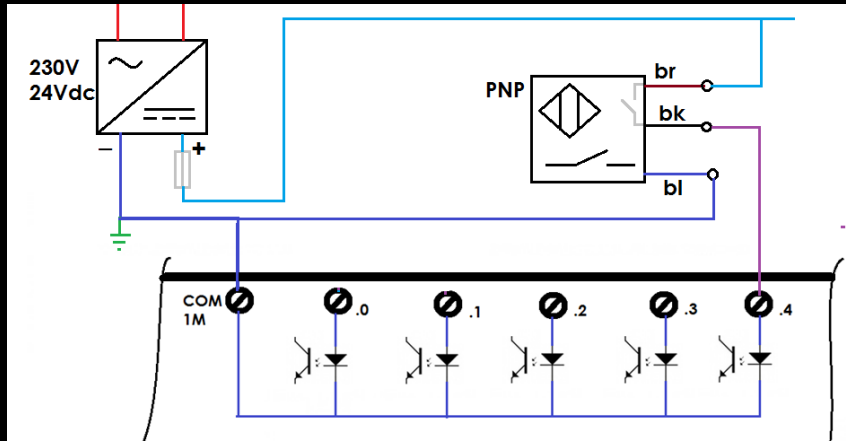
DI: sourcing

- Probeer dit eens zelf op te lossen.



Digitale ingangen: PNP-sensor

DI: sinking



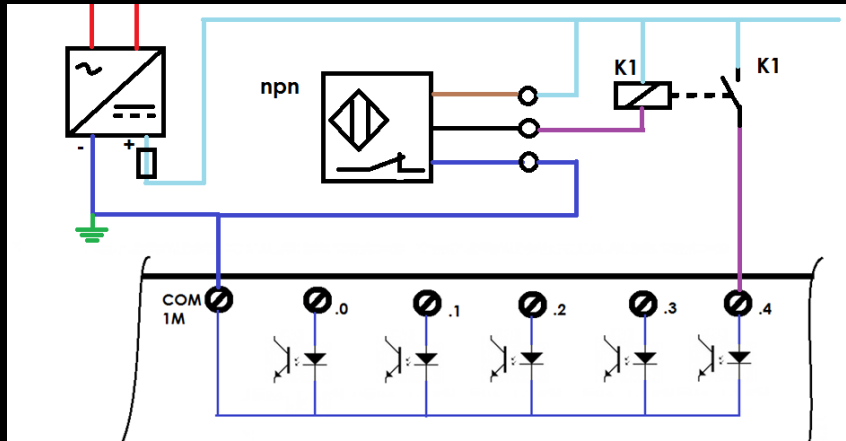
DI: sourcing

- Probeer dit eens zelf op te lossen.



Digitale ingangen: NPN-sensor

DI: sinking



DI: sourcing

- Probeer dit eens zelf op te lossen.



Soorten DI S7-1214C AC/DC/RLY?

- Sinking
- 24V dc
- Max.A

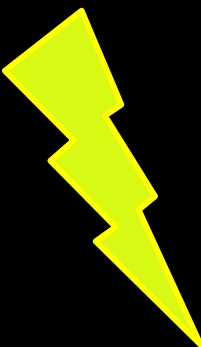


Table A- 24 Specifications for the digital inputs (DI)

Technical data	CPU, SM and SB
Type	<ul style="list-style-type: none">• CPU and SM: IEC Type 1 sink (Sink/Source)• SB 1223: IEC Type 1 sink (Sink only)
Rated voltage	24 VDC at 4 mA, nominal
Continuous permissible voltage	30 VDC, max.
Surge voltage	35 VDC for 0.5 sec.
Logic 1 signal (min.)	15 VDC at 2.5 mA
Logic 0 signal (max.)	5 VDC at 1 mA
Isolation (field side to logic)	500 VAC for 1 minute
Isolation groups	<ul style="list-style-type: none">• CPU: 1• SM 1221 DI 8: 2• SM 1221 DI 16: 4• SB 1223 DI 2: 1• SM 1223: 2
Filter times	0.2, 0.4, 0.8, 1.6, 3.2, 6.4, and 12.8 ms (selectable in groups of 4)
Number of inputs on simultaneously	<ul style="list-style-type: none">• SM 1221 and SM 1223 DI 8: 8• SM 1221 and SM 1223 DI 16: 16• SB 1223 DI 2: 2
Cable length (meters)	<ul style="list-style-type: none">• 500 m shielded, 300 m unshielded• CPU: 50 m shielded for HSC



Digitale uitgangen

- Aansturen van actuatoren
- 2 toestanden mogelijk: aan of uit
 - $DQ = 1?$ \rightarrow Stroom door actuator
 - $DQ = 0?$ \rightarrow geen stroom door actuator



Digitale uitgangen: soorten

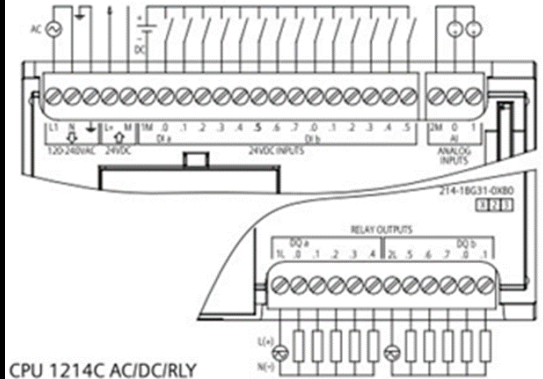
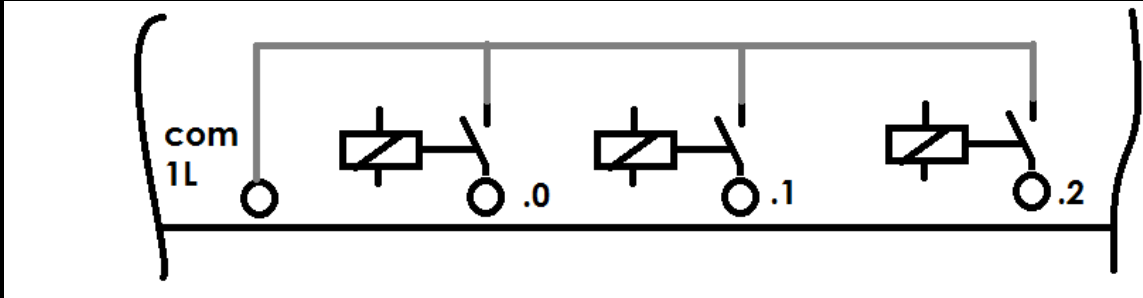
- Relais
- Transistor
- Triac
- Solid State Relay



Digitale uitgangen: soorten

- Relais

- Traag
- Robuust
- P.S. al dan niet gemeensch. klem



CPU 1214C AC/DC/RLY
(6ES7 214-1BG31-0XB0)



Digitale uitgangen: soorten

- Relais
- Transistor
 - Snel
 - gevoelig
- Triac
- Solid State Relay



Digitale uitgangen: soorten

- Relais
- Transistor
- Triac
 - AC
 - snel
- Solid State Relay



Digitale uitgangen: soorten

- Relais
- Transistor
- Triac
- Solid State Relay
 - Zeer snel
 - Gevoelig tegen verkeerde aansluiting!



Digitale uitgangen: S7-1214 AC/DC/RLY

- Relais

- RLY

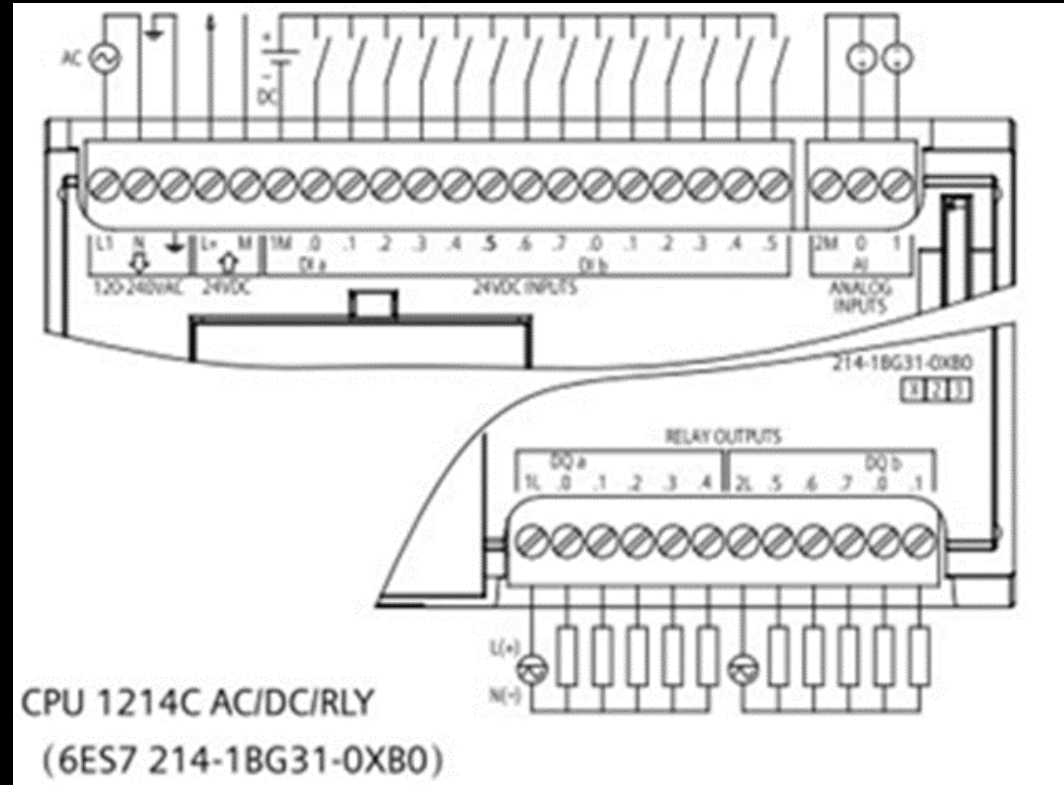
- S7-1214C AC/DC/RLY

- Relais kan verschillende stromen

- én

- verschillende spanningen in- en uitschakelen.

- Hoe zit dit bij de PLC S7-1214C AC/DC/RLY?



Digitale uitgangen: S7-1214 AC/DC/RLY

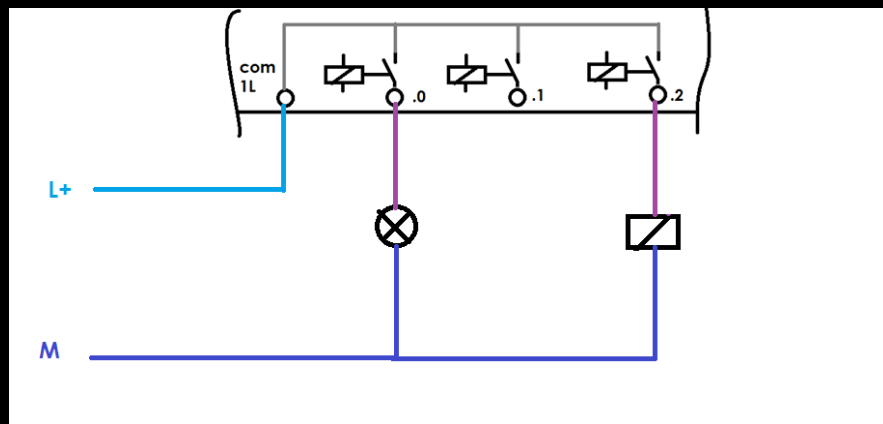


Table A- 27 Specifications for the digital outputs (DQ)

Technical data	Relay (CPU and SM)	24V DC (CPU, SM, and SB)
Type	Relay, dry contact	Solid state - MOSFET (Source)
Voltage range	5 to 30 VDC or 5 to 250 VAC	20.4 to 28.8 VDC
Logic 1 signal at max. current	N/A	20 VDC min.
Logic 0 signal with 10 K Ω load	N/A	CPU: 20 VDC min., 0.1 VDC max. SB: 0.1 VDC max. SM DC: 0.1 VDC max.
Current (max.)	2.0 A	0.5 A
Lamp load	30 W DC / 200 W AC	SB: 5 W
ON state resistance	0.2 Ω max. when new	0.6 Ω max.
OFF state resistance	N/A	N/A
Leakage current per point	N/A	10 μ A max.
Pulse Train Output rate	CPU: N/A ³	CPU: 100 KHz max., 2 Hz min. ⁴ SB: 20 KHz max., 2 Hz min. ⁵
Surge current	7 A with contacts closed	CPU: 8 A for 100 ms max. SB: 5 A for 100 ms max. SM: 8 A for 100 ms max.
Overload protection	No	No
Isolation (field side to logic)	Coil to contact: 1500 VAC for 1 minute Coil to logic: None	500 VAC for 1 minute

ADRESSERING? REGISTER?

GROOTTE & SOORTEN DATA TYPES?



ADRESSERING: BASIS

Grootte van een datatype/waarde:

Bit Byte Word DWord

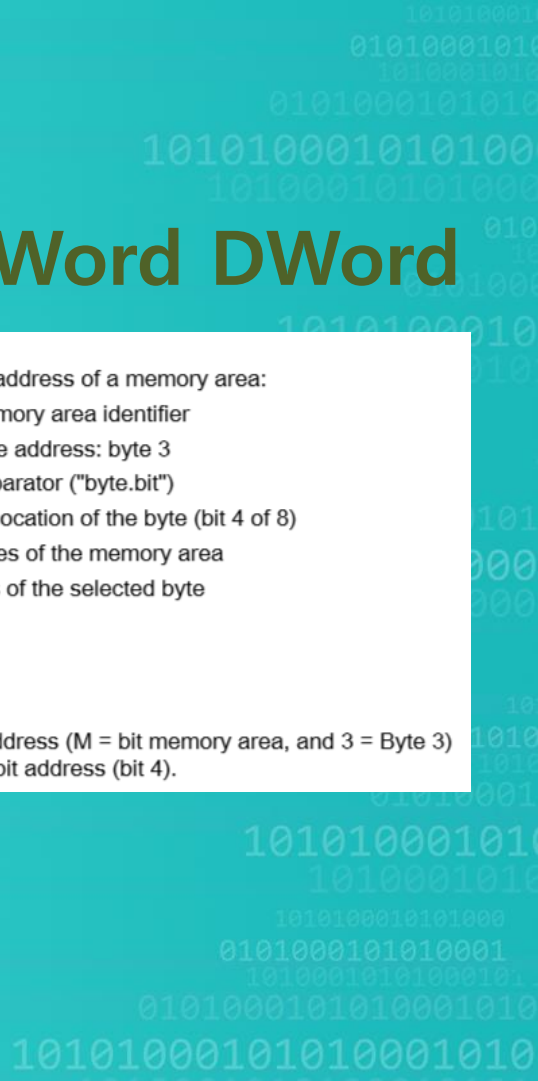
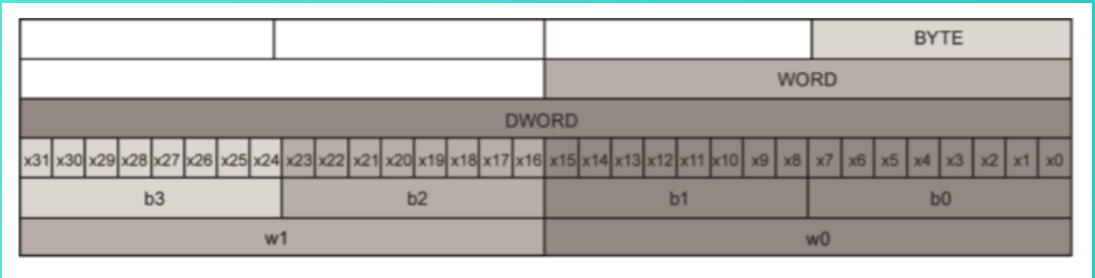
- | | |
|---------------------------------|--|
| Bit and bit-sequence data types | <ul style="list-style-type: none"> • Bool is a Boolean or bit value. • Byte is an 8-bit byte value. • Word is a 16-bit value. • DWord is a 32-bit double-word value. |
|---------------------------------|--|

M 3 . 4
 (A) (B) (C) (D)

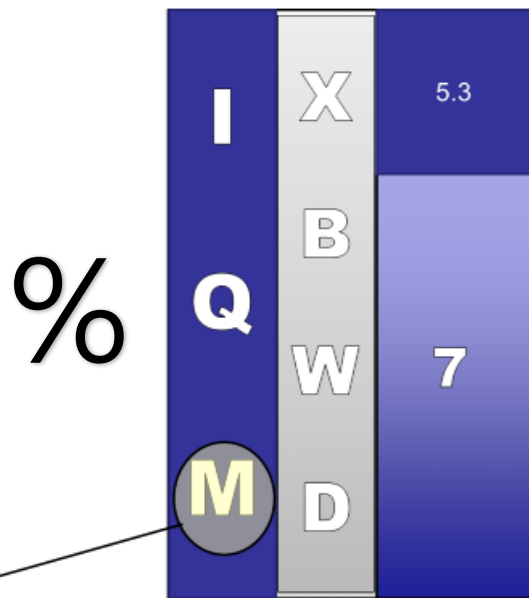
Absolute address of a memory area:

- A Memory area identifier
- B Byte address: byte 3
- C Separator ("byte.bit")
- D Bit location of the byte (bit 4 of 8)
- E Bytes of the memory area
- F Bits of the selected byte

In the example, the memory area and byte address (M = bit memory area, and 3 = Byte 3) are followed by a period (".") to separate the bit address (bit 4).



Terugkerende opbouw variabelen



Omron: H of HR
Hulpregister ipv merkerregister



Register

In een PLC zitten er een aantal 'spreadsheets'.

Elk heeft een eigen naam:

- Ingangsregister I
- Uitgangsregister Q
- Merkerregister M
- Tellers C
- Timers T
- ...

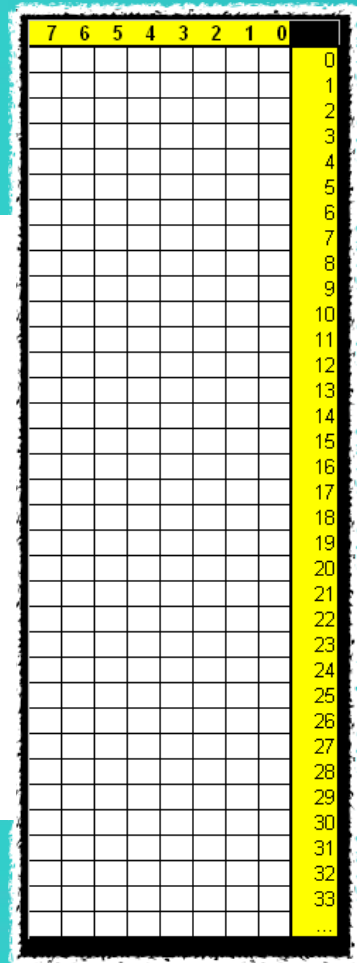
7	6	5	4	3	2	1	0
							0
							1
							2
							3
							4
							5
							6
							7
							8
							9
							10
							11
							12
							13
							14
							15
							16
							17
							18
							19
							20
							21
							22
							23
							24
							25
							26
							27
							28
							29
							30
							31
							32
							33
							...



Lengte variabele?

LENGTES	<u>lengte in REGISTER</u>	<u>Voorbeeld</u>
- BIT:	1 cel	M1.5
- BYTE:	1 rij	MB3
- WOORD:	2 rijen	MW16
- DUBBELWOORD:	4 rijen	MD10

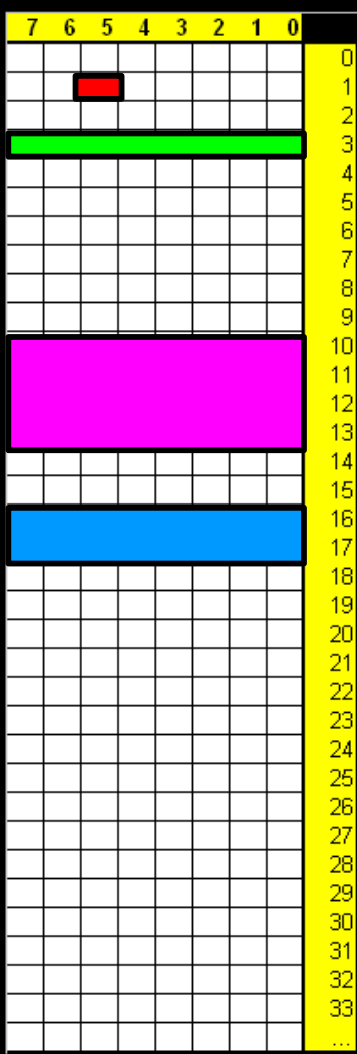
Opdracht stel de voorbeelden voor in het merkerregister op volgende slide.



Oefening: oplossing

- Voorbeeld

- BIT: M1.5
- BYTE: MB3
- WOORD: MW16
- DUBBELWOORD: MD10



OEFENBLAD

7	6	5	4	3	2	1	0	
								0
								1
								2
								3
								4
								5
								6
								7
								8
								9
								10
								11
								12
								13
								14
								15
								16
								17
								18
								19
								20
								21
								22
								23
								24
								25
								26
								27
								28
								29
								30
								31
								32
								33
								...

7	6	5	4	3	2	1	0	
								0
								1
								2
								3
								4
								5
								6
								7
								8
								9
								10
								11
								12
								13
								14
								15
								16
								17
								18
								19
								20
								21
								22
								23
								24
								25
								26
								27
								28
								29
								30
								31
								32
								33
								...

7	6	5	4	3	2	1	0	
								0
								1
								2
								3
								4
								5
								6
								7
								8
								9
								10
								11
								12
								13
								14
								15
								16
								17
								18
								19
								20
								21
								22
								23
								24
								25
								26
								27
								28
								29
								30
								31
								32
								33
								...

101010001
01010001010
1010001010
0101000101010
0001010100
0001010100
010
0101000
1010100010
10100010
0101
10101000
101000
10
0101
1010
01010001
01010001010
1010001010
010100010101000
000101010001
100010101000101
101010001010
1010001010

BASISPROGRAMMA'S

- 1. Bit logica
- 2. Set/Reset
- 3. Timers
- 4. Flankdetectie
- 5. Oefeningen



ProgrammerMETHODEN

- 🌀 LAD: Ladder
- 🌀 FBD: Functieblok Diagramma
- 🌀 STL: Structured Text Language
- 🌀 IL: Instruction list
- 🌀 SFC: Sequential Function Chart



Bit Logica:



Contacts test the binary status of the bit, with the result being "power flow" if on (1) or "no power flow" if off (0).

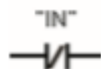
The state of the coil reflects the status of the preceding logic.

If you use a coil with the same address in more than one program location, the result of the last calculation in the user program determines the status of the value that is written to the physical output during the updating of the outputs.

Contact Open



Contact Closed



The Normally Open contact is closed (ON) when the assigned bit value is equal to 1.

The Normally Closed contact is closed (ON) when the assigned bit value is equal to 0.



Bits bekijken

1

Het contact is open



Spanningsniveau?

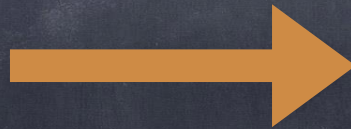
0

Het contact is gesloten



DI: n.o. vs n.c.

- Vraag 1: type no of nc
- Vraag 2: wordt de schakelaar bediend?
- Vraag 3: komt er spanning toe?
- Symbool afh van 3



Oefening:

hoe bediende startknop voorstellen?

- Startknop = n.o.
- N.o. Is bediend
- Spanning aan input



Oefening: hoe bediende stopknop vo orstellen?

- Stopknop = n.c.
- N.c. Is bediend
- Geen spanning aan input



Oefening: hoe niet bediende startknop voors tellen?

- Startknop = n.o.
- N.o. is NIET bediend
- Geen spanning aan input



Oefening: hoe niet bediende stopknop voors tellen?

- Stopknop = n.c.
- n.c. is NIET bediend
- Spanning aan input



	Bediend?	Spanning?	Symbol?
--	----------	-----------	---------

N.O.

Neen

Neen



N.O.

Ja

Ja



N.C.

Ja

Neen



N.C.

Neen

Ja



Bitlogica uitgangen Q

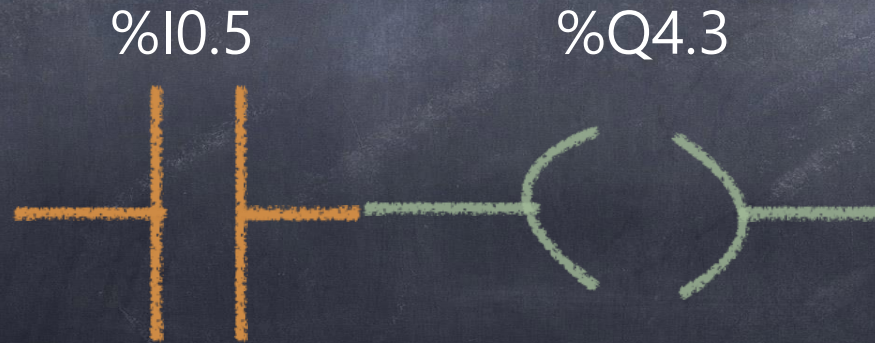
- Q slechts EENMAAL voorkomen in één netwerk in het gans programma!!!
- Tenzij set- & resetfunctie.
- Voorstellen:



Bitlogica: Oef.

- Oefeningen op bitlogica
- @ LAD

- 1-01: als men op een n.o. drukknop duwt dan brandt er een lampje. Bij het loslaten gaat het lampje uit. De drukknop is aangesloten op input 0.5; het lampje op output 4.3.



- 1-02. Een lampje (Q4.3) brandt als drukknop I0.2 (n.c.) ingedrukt is.

%I0.2

%Q4.3

-
- 1-03. Een lampje (Q4.3) brandt als drukknop I0.2 (n.c.) niet ingedrukt is.

- 1-04: zolang men niet op een n.o. drukknop duwt dan brandt er een lampje. Bij het indrukken gaat het lampje uit, bij loslaten gaat het lampje terug aan. De drukknop is aangesloten op input 0.5; het lampje op output 4.3.

- 1-05: Twee n.o. drukknoppen (I0.5; I0.4) moeten samen ingedrukt worden om een lampje Q4.4 te doen branden.

- 1-06: als een van de twee n.o. drukknoppen (I0.5; I0.4) ingedrukt wordt, dan zal een lampje Q4.4 branden.

- 1-07: Lampje Q4.4 zal pas branden wanneer ofwel drukknop 1 (I0.4) ofwel drukknop 2 (I0.5) ingedrukt is. Het lampje brandt niet als beide drukknoppen dezelfde toestand hebben.

- 1-08: een eindeloopschakelaar I1.0 (n.o.) wordt bekrachtigd door een palet. Op dat moment moet er een oranje lamp (Q4.7) branden.

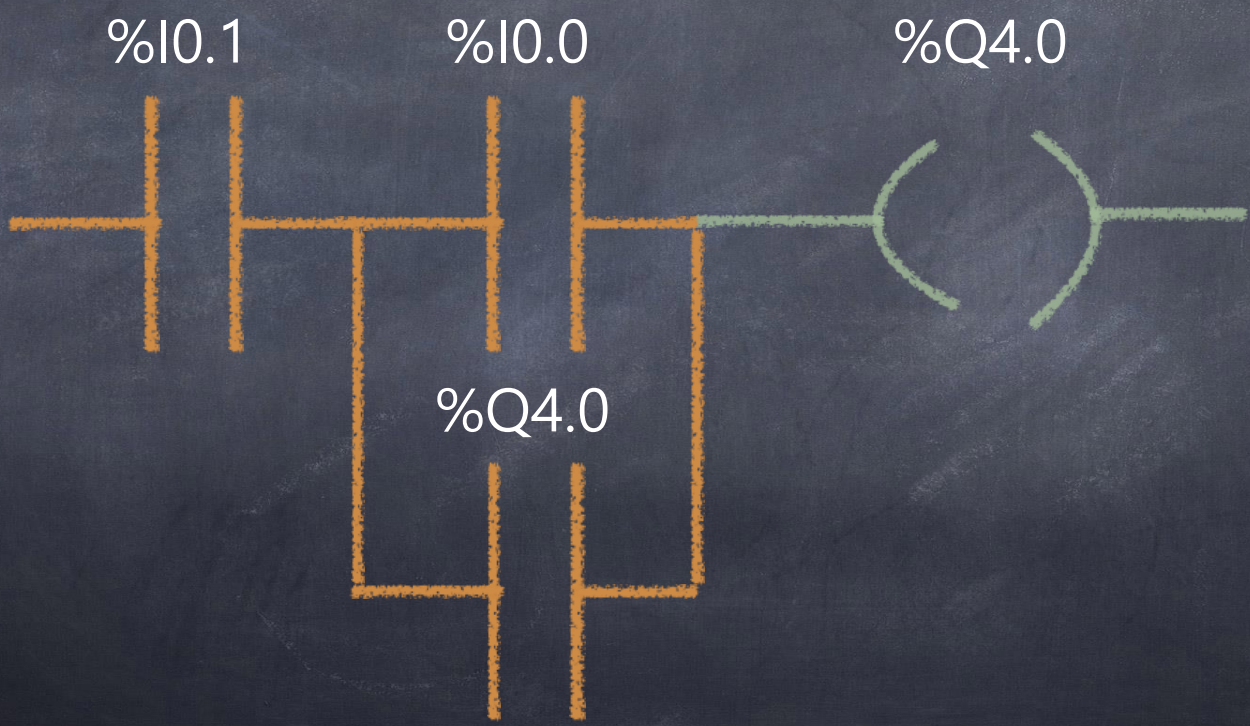
Bitlogica: behoud van toestand

- Hoe kan men een toestand onthouden zonder een drukknop te blijven indrukken?
- Uitgewerkt voorbeeld: duwen op start (n.o.; I0.0) doet lamp (q4.0) branden. Die gaat pas uit bij drukken op stop (n.c.; I0.1)

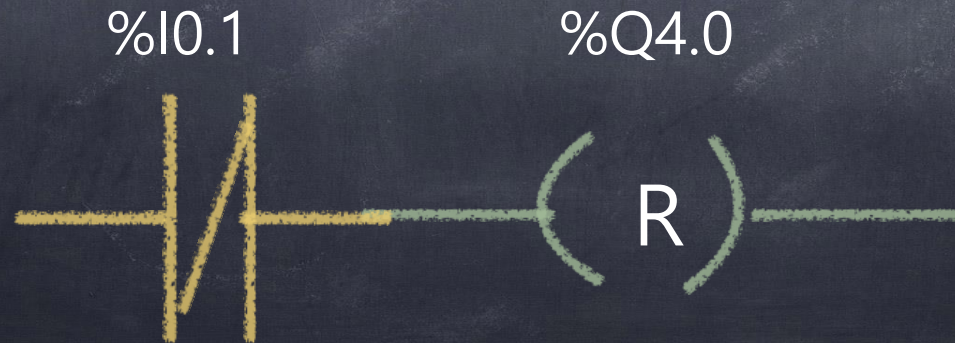
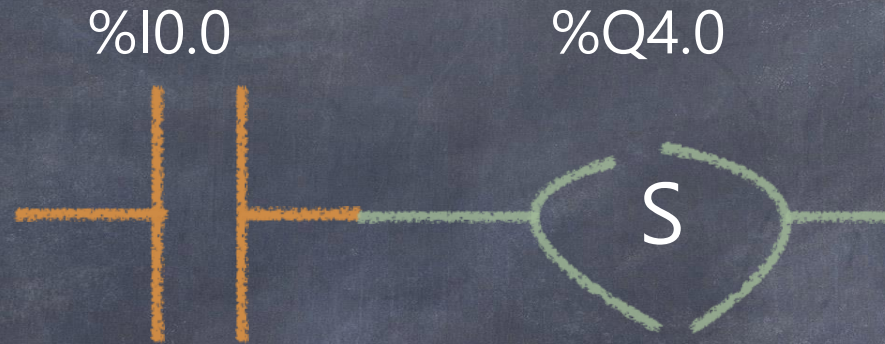
Bitlogica: behoud van toestand

- OPL 1: Overneemcontact
- OPL 2: Set & Reset

Opl 1: overneemcontact



Opl 2:
Set/Reset-functie



Oefeningen

Hierna volgen een aantal oefeningen
die u individueel probeert te maken...

- 2-01:

- Een relais (Q10.0) stuurt een lamp aan. Als de aan-knop (n.o.; IO.0) ingedrukt wordt, dan brandt het lampje. Wanneer er op de uit-knop (n.o.; IO.1) gedrukt wordt, dan gaat het lampje uit.

- 2-02:

- Een enkelwerkende cilinder wordt door een monostabiel 4/2-ventiel aangestuurd. Het ventiel wordt bekrachtigd bij het indrukken van de startknop (IO.0). Wanneer de cilinder de eindelooop S2 (IO.3; n.c.) nadert, dan wordt het ventiel uitgeschakeld en keert de cilinder terug.

- 2-03:

- Een drukknop S1 (I0.4) wordt gebruikt om een lamp aan te zetten. Na 10 seconden wordt de lamp uitgezet.

- 2-04:

- Een enkelwerkende cilinder wordt door een monostabiel 4/2-ventiel aangestuurd. Het ventiel wordt bekrachtigd bij het indrukken van de startknop (I0.0). Wanneer de cilinder de eindloop S2 (I0.3; n.c.) nadert, dan wordt het ventiel uitgeschakeld en keert de cilinder terug. Bij indrukken van stop moet er direct gestopt worden.

- 2-05:

- Een startknop voor ster (3f ASM; Y; I0.2; n.o.) zorgt er voor dat de motor opstart in Y. Een startknop voor driehoek (D, I0.3; n.o.) laat de motor in driehoek starten. De motor mag enkel maar starten als ze stilstand. Een stopknop (I0.1; n.c.) schakelt in beide gevallen de motor uit.

- 2-06:

- Praktisch probleem: een motor wordt aangedreven via een relais verbonden met een PLC; U kan niet direct aan het programma. Ingang I0.0 is voorzien als een n.o.-contact, maar de drukknop is helaas stuk. De machine moet dringend vermaakt worden, maar in je werkbak zitten er enkel een n.c.-schakelaar en een relais van Finder (zie blauwe model in klas). Hoe los je het probleem op?

Noot:

PLC-programmacyclus

- Zet men de PLC in run dan draait het programma MAIN
-
- Het programma wordt uitgevoerd, rij per rij
- Op het einde van de rijen gekomen?
 - opnieuw beginnen
 - én telkens opnieuw en opnieuw en ...

Noot:

PLC-programmacyclus

- De juiste manier van programmeren:
- Er is een MAIN dat uitgevoerd wordt.
- In de MAIN verwijst je naar de diverse onderdelen (FB, FC,...) die in jouw opdracht moeten draaien.

Bitlogica flankdetectie

- Wat?
 - Detectie van de verandering van de input
 - Positieve vs Negatieve flankdetectie

Oefeningen:

- Voorbeeld: een lamp gaat branden (%Q2.4) als de no drukknop (%I4.2) ingedrukt wordt.



- 03-01: Een motor draait rechts tot de eindeloopschakelaar bereikt wordt. Op dat moment wacht de motor 5 sec alvorens naar links te draaien.
- 03-02: Een lamp gaat aan of uit afhankelijk van de impuls van de drukknop.
- 03-03: Start met behulp van een drukknop een motor, en stop de motor met dezelfde drukknop.

- 03-04: Een ASM draait links-rechts. De motor drijft een stang aan met een karretje. Dit karretje beweegt tussen de linkerkant en de rechterkant, op elk uiteinde zit er een sensor die zorgt dat de motor stopt en draait van kant.

- Start: I0.0 n.o.

- Stop: I0.1 n.c.

- Sensor Links: n.o. I0.4

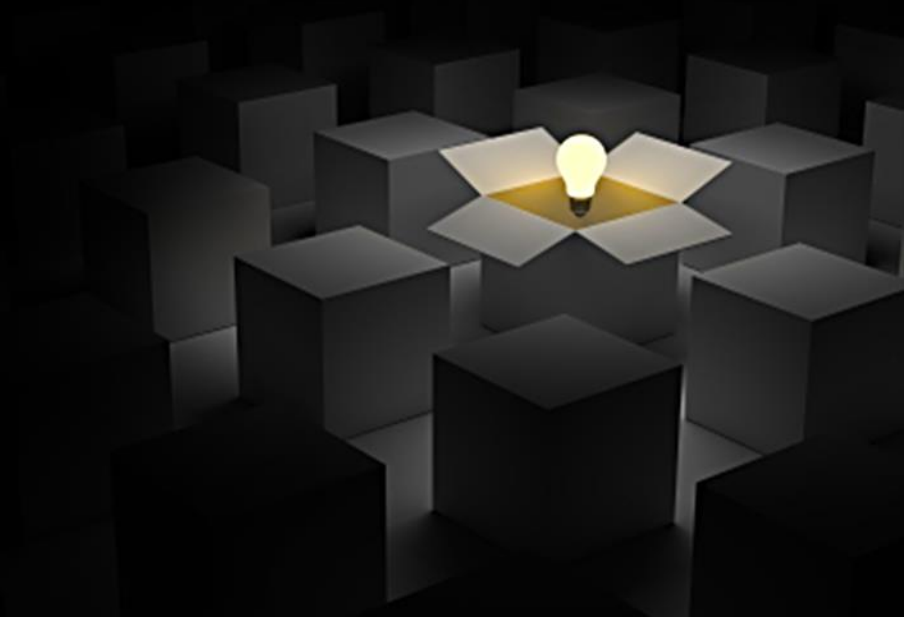
- Sensor Rechts: n.o. I0.5

- Relais links: Q0.0

- Relais Rechts: Q0.1

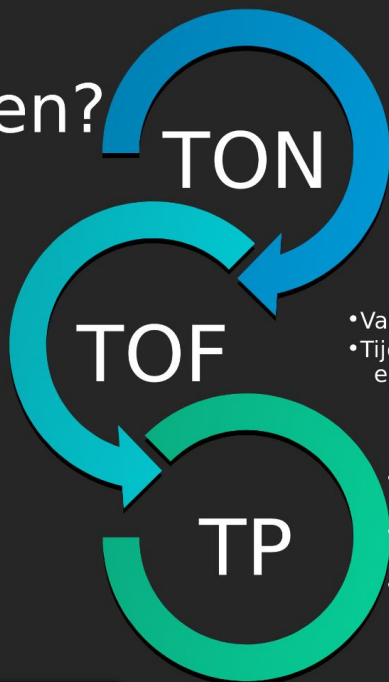


Timers



Siemens TIA v11

Welke timer kiezen?



- AAN na ingestelde tijd
- Wegvallen enable, uitgang laag

- Valt UIT na ingestelde tijd
- Tijd begint te tellen bij wegvallen enable

- PULS AAN gedurende ingestelde tijd
- Tijd begint te tellen bij hoog worden input.
- Valt uit na de ingestelde tijd - ongeacht status enable



TIME?

Data Type TIME


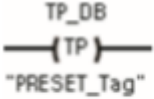
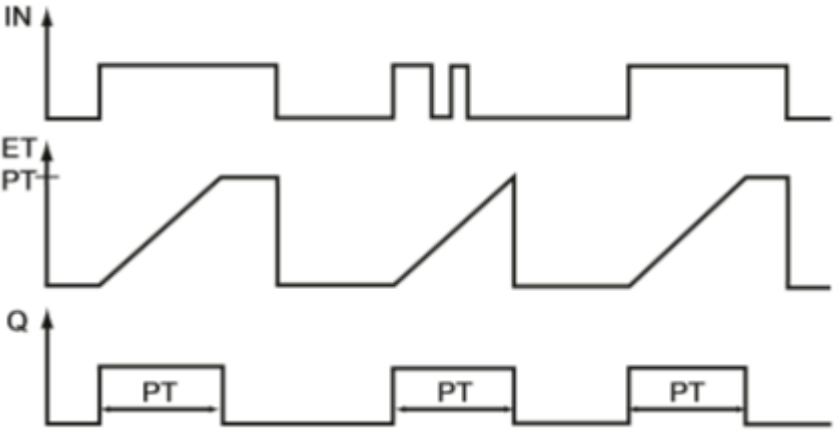
- 32 bits
 - Dint
 - Gaat van:
 - -2 147 483 648 ms
- Tot
- 2 147 483 647 ms

Data type	Size	Valid number ranges ¹
TIME	32 bits, stored as DInt data	T#-24d_20h_31m_23s_648ms to T#24d_20h_31m_23s_647ms Stored as -2,147,483,648 ms to +2,147,483,647 ms

¹ The negative range of the TIME data type shown above cannot be used with the timer instructions. Negative PT(preset time) values are set to zero when the timer instruction is executed. ET(elapsed time) is always a positive value.



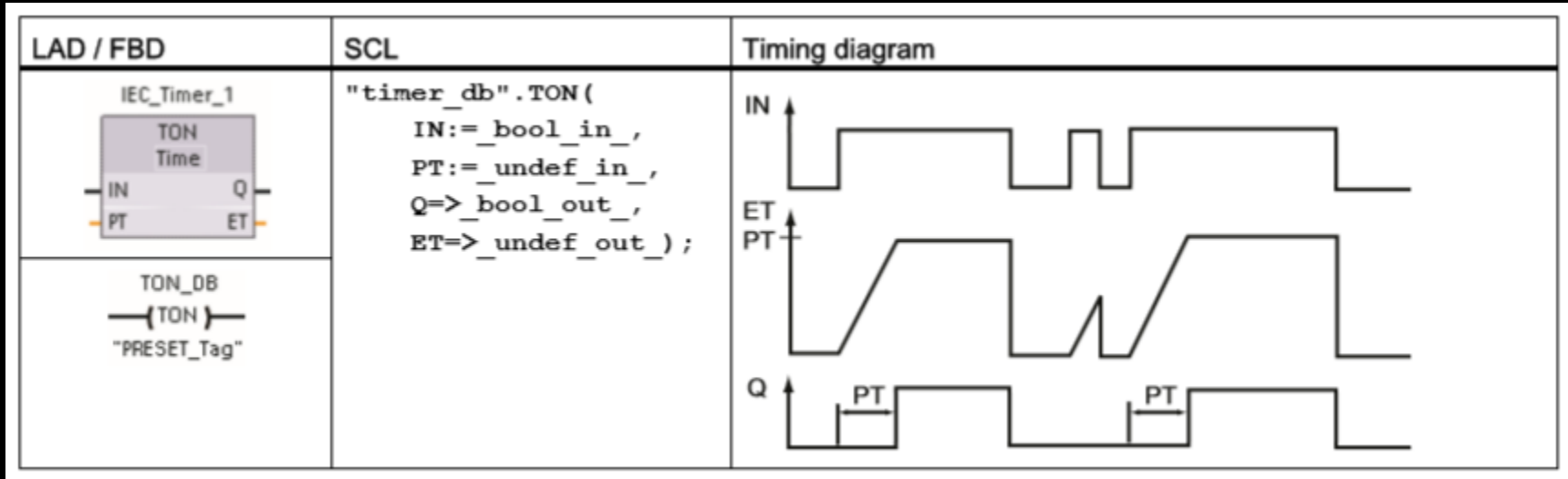
TP = Pulse Timer

LAD / FBD	SCL	Timing diagram
<p>IEC_Timer_0</p>  <p>TP_DB</p> 	<pre>"timer_db".TP(IN:=_bool_in_, PT:=_undef_in_, Q=>_bool_out_, ET=>_undef_out_);</pre>	

→ Leg in eigen woorden uit wat deze functie doet.



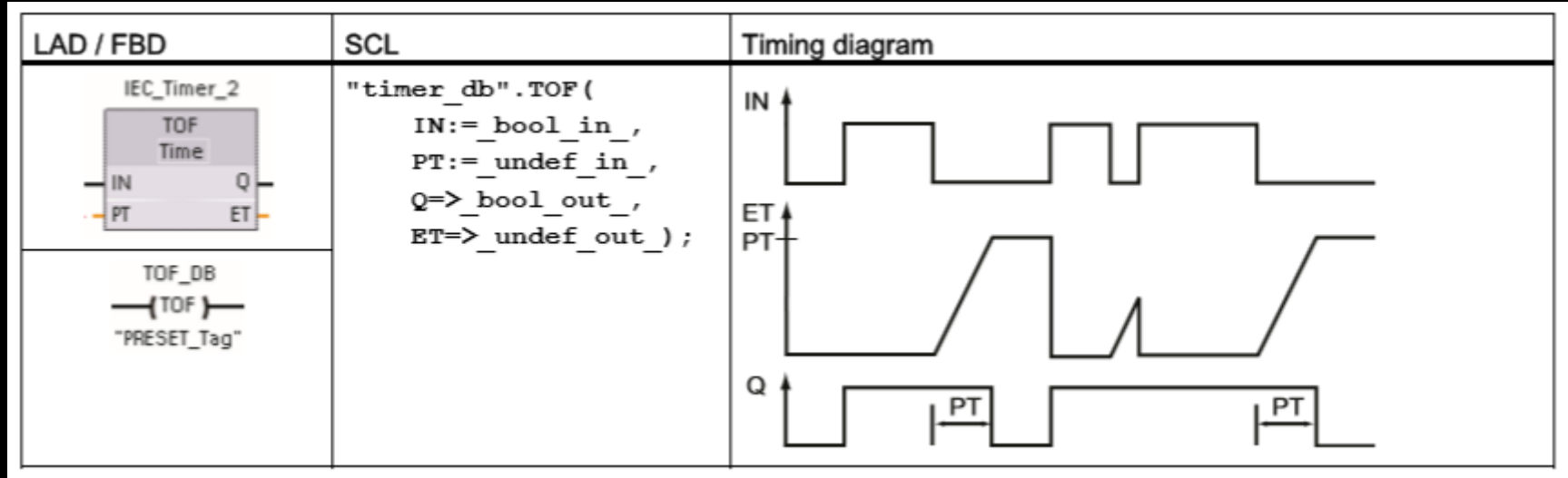
TON: on-delay timer → vertraagd inschakelend



→ Leg in eigen woorden uit wat deze functie doet.



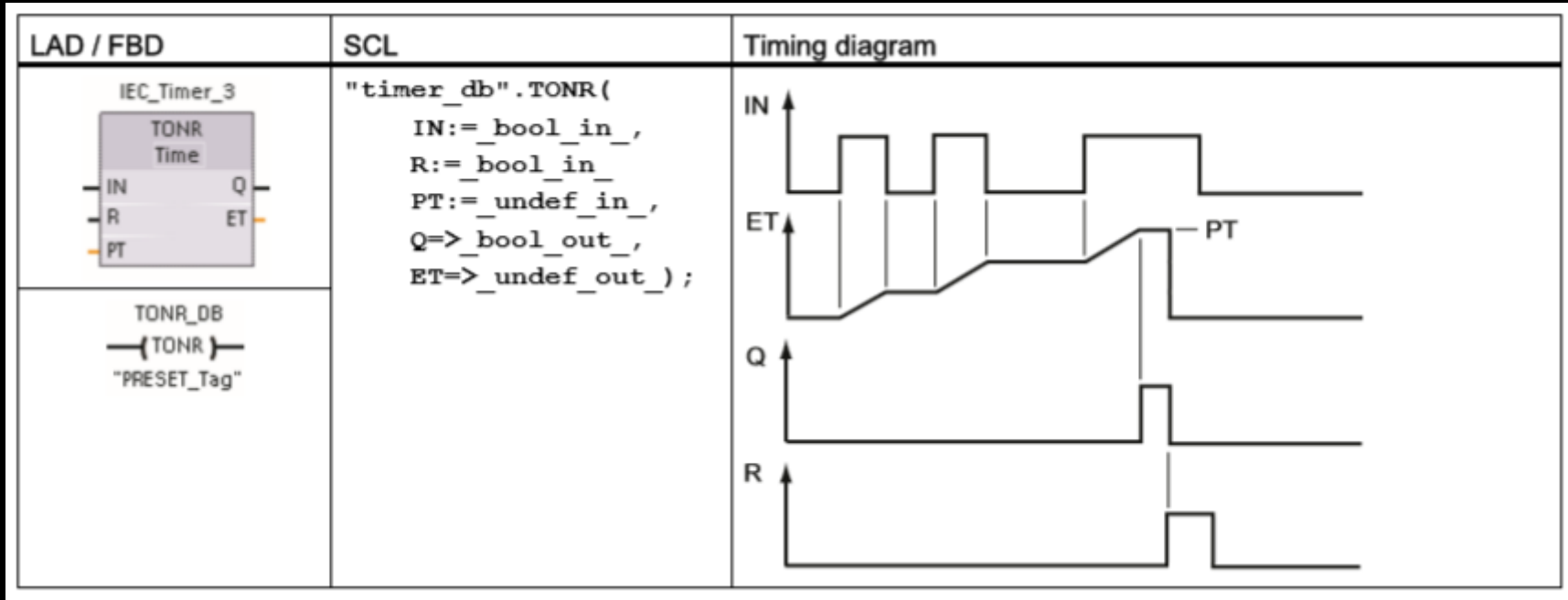
TOF = OFF-delay timer → vertraagd uitschakelend



→ Leg in eigen woorden uit wat deze functie doet.



TONR = ON-delay retentive timer



→ Leg in eigen woorden uit wat deze functie doet.



COUNTERS = TELLERS

Counters

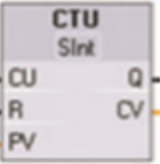
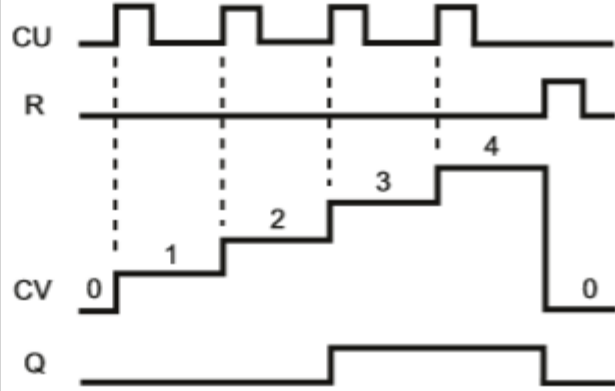
You use the counter instructions to count internal program events and external process events.

- The "count up" counter (CTU) counts up by 1 when the value of the input parameter CU changes from 0 to 1.
- The "count down" counter (CTD) counts down by 1 when the value of input parameter CD changes from 1 to 0.
- The "count up and down" counter (CTUD) counts up or down by 1 on the 0 to 1 transition of the count up (CU) or count down (CD) inputs.

S7-1200 also provides high-speed counters (Page 111) (HSC) for counting events that occur faster than the OB execution rate.



CTU: optellen

LAD / FBD	SCL	Operation
<p>"Counter name"</p>  <p>The LAD symbol shows a rectangular box labeled "CTU" with "Sint" below it. On the left side, there are three input terminals: "CU" (top), "R" (middle), and "PV" (bottom). On the right side, there are two output terminals: "Q" (top) and "CV" (bottom).</p>	<pre>"ctu_db".CTU(CU:=_bool_in, R:=_bool_in, PV:=_undef_in, Q=>_bool_out, CV=>_undef_out);</pre>	 <p>The timing diagram illustrates the counter's behavior. The CU input has four rising edges. The R input has a single pulse. The CV output is a staircase function that increases from 0 to 4 with each rising edge of CU, and resets to 0 when R is active. The Q output is a pulse that occurs when CV reaches 4.</p>

- Merk de flankwerking op van de input
- Leg in eigen woorden de werking van deze functie uit.



Getalformaten in een PLC?

- Max grootte begrensd door #bits
- BCD vs Hexadecimaal vs binair
- Cijferformaten (real, int, time, date)
- Tekens (ASCII)



Getalformaten in een PLC?

- Decimaal?

$$\text{B.v. } 23 = 2 \times 10 + 3 \times 1$$

- BCD (binary coded decimal)?
- Hexadecimaal?



Getalformaten in een PLC?

- Decimaal?
- BCD (binary coded decimal)?
- Hexadecimaal?
 - Bij codering van decimaal of BCD is er dataverlies.
 - Oplossing? Hexadecimaal = 16

d	h	opslag
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111



Getalformaten in een PLC?

- Decimaal?
- BCD (binary coded decimal)?
- Hexadecimaal?

– B.v. $23 = 1 \times 16 + 7 \times 1 = 17_h = 0001\ 0111_h$

d	h	opslag
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111



Getalformaten in een PLC?

- Hexadecimaal?

- B.v. 23 = $1 \times 16 + 7 \times 1 = 17_h = 0001\ 0111_h$

- B.v. 44 = $2 \times 16 + 12 \times 1 = 2 \times 16 + C \times 1$
= $2C_h = 0010\ 1100_h$

- B.v. 174 = $10 \times 16 + 14 \times 1 = AE_h$

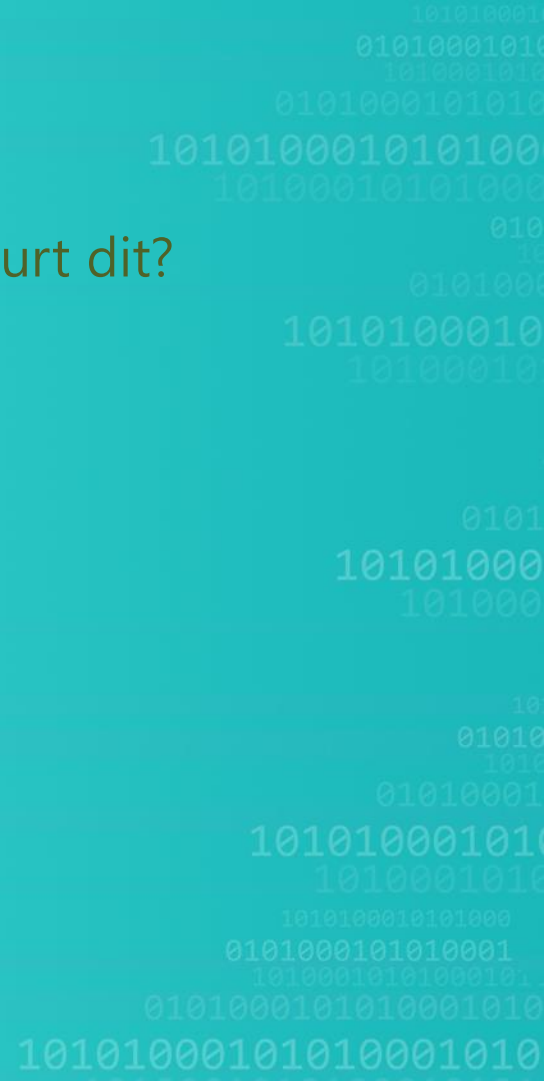
d	h	opslag
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111



Data types

Wat bepaalt de grootte in het geheugen? Hoe gebeurt dit?

Integer data types	<ul style="list-style-type: none">• USInt (unsigned 8-bit integer) and SInt (signed 8-bit integer) are "short" integers (8 bits or 1 byte of memory) that can be signed or unsigned.• UInt (unsigned 16-bit integer) and Int (signed 16-bit integer) are integers (16 bits or 1 word of memory) that can be signed or unsigned.• UDInt (unsigned 32-bit integer) and DInt (signed 32-bit integer) are double integers (32 bits or 1 double-word of memory) that can be signed or unsigned.
Real number data types	<ul style="list-style-type: none">• Real is a 32-bit Real number or floating-point value.• LReal is a 64-bit Real number or floating-point value.



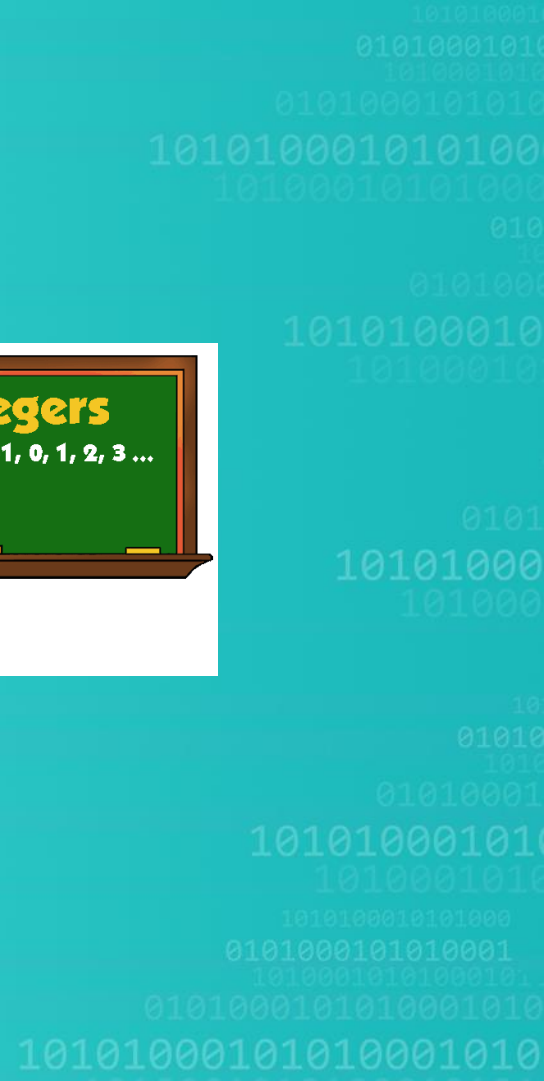
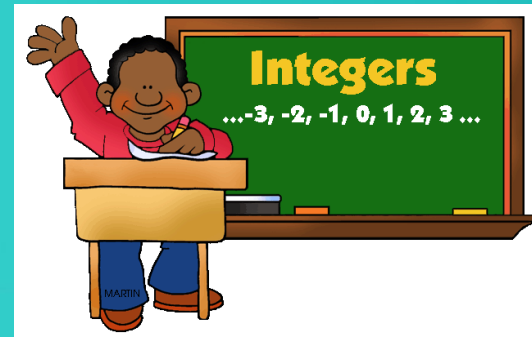
Data Types?

INTEGER vs UNSIGNED INTEGER vs REAL

Natuurlijke getallen = UNSIGNED INTEGER

Gehele getallen = SIGNED INTEGER

Reële getallen = REAL



UNSIGNED INTEGER ($\forall x: x \in N$)

SOORTEN NATUURLIJKE GETALLEN? Afh. van aantal bits!!!!

	# bits?	Min. (bin)	Min. (dec.)	Max. (bin)	Max. (dec.)
USINT	8	0000 0000	0	1111 1111	$2^8 - 1 = 255$
UINT	16	0000 0000 0000 0000	0	1111 1111 1111 1111	$2^8 - 1 =$
UDINT	32	0000 0000 0000 0000 0000 0000 0000 0000	0	1111 1111 1111 1111 1111 1111 1111 1111	$2^{32} - 1 =$
ULINT	64	0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000	0	1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111	$2^{64} - 1 = \dots$ (schaakbord)

SIGNED INTEGER ($\forall x: x \in \mathbb{Z}$)

Tekenbit!!! 1: minteken (-)

	# bits?	Min. (bin)	Min. (dec.)	Max. (bin)	Max. (dec.)
SINT	8	1xxx xxxx	0	0111 1111	$2^8 - 1 = 255$
UINT	16	xxxx xxxx 1xxx xxxx	0	1111 1111 0111 1111	$2^8 - 1 =$
UDINT	32	xxxx xxxx xxxx xxxx xxxx xxxx 1xxx xxxx	0	1111 1111 1111 1111 1111 1111 0111 1111	$2^{32} - 1 =$
ULINT	64	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx 1xxx xxxx	0	1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111	$2^{64} - 1 = \dots$ (schaakbord)

SIGNED INTEGERS

Waarom die x na de tekenbit?

Stel bij SINT gewoon en alleen het tekenbit en de andere 7 bits geïnverteerd:

Positieve getallen:

$$0\ 000\ 0000 = 0$$

$$0\ 111\ 1111 = 2^7 - 1 = 127$$

128 getallen

Negatieve getallen:

$$1\ 000\ 0000 = 0$$

$$1\ 111\ 1111 = -127$$

MERK OP: ER IS TWEE MAAL NUL!!!

→ Hoewel 256 posities zijn er maar 255 getallen mogelijk!

→ Bewerkingen op schaal van de IC's zijn er veel bewerkingen nodig (tragere P(L)C)



SIGNED INTEGERS

Waarom die x na de tekenbit? → 1st complement?

Voorstellen van -10 (SINT)?

10 → binair?

→ 0000 1010

→ complement? (0 wordt 1; 1 wordt 0)

→ 1111 0101

→ eerste teken is 1 (dus negatief getal)

→ nadeel 1st complement?

→ nog altijd een getal minder dat maximaal mogelijk!

→ getal 0 → 1111 1111 → -0 → een offset van -1



SIGNED INTEGERS

Waarom die x na de tekenbit? → 2nd complement?

Voorstellen van -10 (SINT)?

10 → binair?

→ 0000 1010

→ 1st complement? (0 wordt 1; 1 wordt 0)

→ 1111 0101

→ 1 optellen?

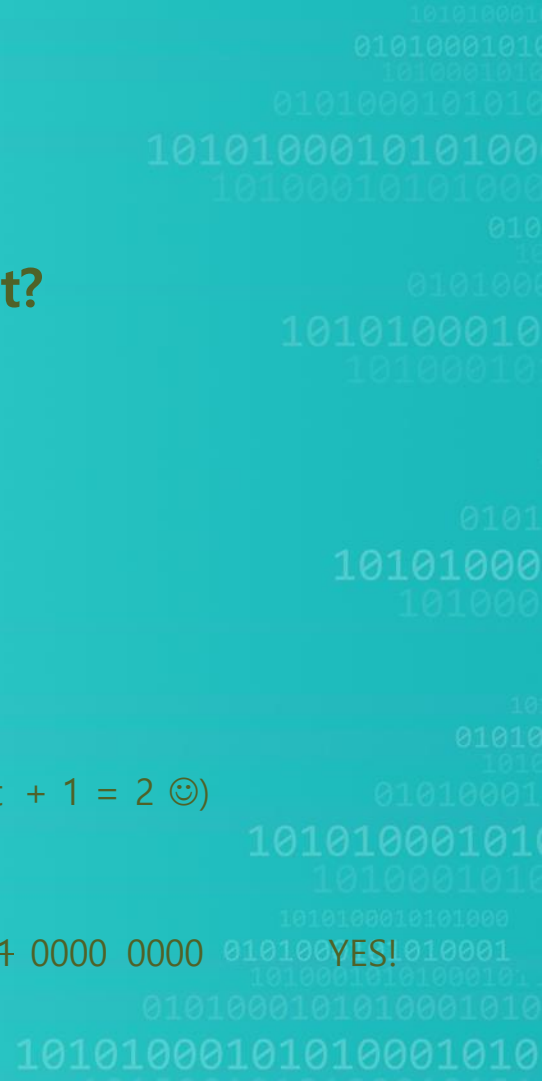
→ 1111 0110

→ 2nd complement (1st complement + 1 = 2 ☺)

→ getal meer!!!

→ 0 → 0000 0000 → 1111 1111 → + 0000 0000

→ direct optellen mogelijk!



SIGNED INTEGERS

Waarom die x na de tekenbit? → 2nd complement?

Voorstellen van -25 (SINT)?

25 → binair?

→ 0001 1001

→ 1st complement? (0 wordt 1; 1 wordt 0)

→ 1110 0110

→ 1 optellen?

→ 1110 0111

NOT POORTEN

ADD POORTEN zijn de makkelijkste in een PC!!! → werken het snelst! → hoe meer toepassingen van + en NOT hoe beter.

REAL

Reële getallen?

Tekenbit(s)

Basisgetal

Exponent van 10

b.v. -23745×10^{-5}



Data types

Wat bepaalt de grootte in het geheugen? Hoe gebeurt dit?

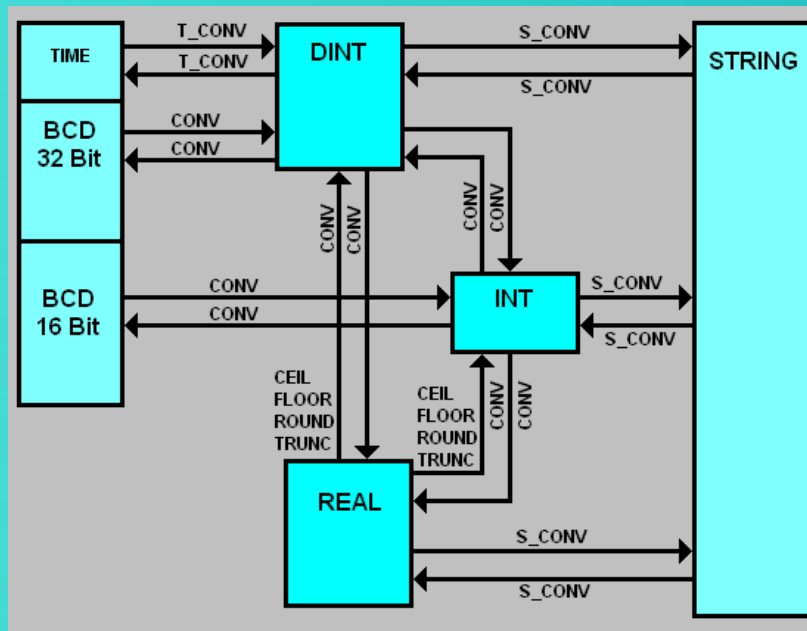
Data types	Description
Date and time data types	<ul style="list-style-type: none">• Date is a 16-bit date value (similar to a UInt) that contains the number of days since January 1, 1990. The maximum date value is 65378 (16#FF62), which corresponds to December 31, 2168. All possible Date values are valid.• DTL (date and time long) is a structure of 12 bytes that saves information on date and time in a predefined structure.<ul style="list-style-type: none">– Year (UInt): 1970 to 2554– Month (UInt): 1 to 12– Day (UInt): 1 to 31– Weekday (UInt): 1 (Sunday) to 7 (Saturday)– Hours (UInt): 0 to 23– Minutes (UInt): 0 to 59– Seconds (UInt): 0 to 59– Nanoseconds (UDInt): 0 to 999999999• Time is a 32-bit IEC time value (similar to a Dint) that stores the number of milliseconds (from 0 to 24 days 20 hours 31 minutes 23 seconds and 647 ms). All possible Time values are valid. Time values can be used for calculations, and negative times are possible.• TOD (time of day) is a 32-bit time-of-day value (similar to a Dint) that contains the number of milliseconds since midnight (from 0 to 86399999).
Character and string data types	<ul style="list-style-type: none">• Char is an 8-bit single character.• String is a variable-length string of up to 254 characters.

Data types

Wat bepaalt de grootte in het geheugen? Hoe gebeurt dit?

Array and structure data types	<ul style="list-style-type: none">• Array contains multiple elements of the same data type. Arrays can be created in the block interface editors for OB, FC, FB, and DB. You cannot create an array in the PLC tags editor.• Struct defines a structure of data consisting of other data types. The Struct data type can be used to handle a group of related process data as a single data unit. You declare the name and internal data structure for the Struct data type in the data block editor or a block interface editor. <p>Arrays and structures can also be assembled into a larger structure. A structure can be nested up to eight levels deep. For example, you can create a structure of structures that contain arrays.</p>
PLC data types	<p>PLC Data type is a user-defined data structure that defines a custom data structure that you can use multiple times in your program. When you create a PLC Data type, the new PLC Data type appears in the data type selector drop drop-lists in the DB editor and code block interface editor.</p> <p>PLC Data types can be used directly as a data type in a code block interface or in data blocks.</p> <p>PLC Data types can be used as a template for the creation of multiple global data blocks that use the same data structure.</p>
Pointer data types	<ul style="list-style-type: none">• Pointer provides an indirect reference to the address of a tag. It occupies 6 bytes (48 bits) in memory and can include the following information to a variable: DB number (or 0 if the data is not stored in a DB), memory area in the CPU, and the memory address.• Any provides an indirect reference to the beginning of a data area and identifies its length. The Any pointer uses 10 bytes in memory and can include the following information: Data type of the data elements, number of data elements, memory area or DB number, and the "Byte.Bit" starting address of the data.• Variant provides an indirect reference to tags of different data types or parameters. The Variant pointer recognizes structures and individual structural components. The Variant does not occupy any space in memory.

Omzetten getallen in TIA? (belangrijk bij gebruik HMI)



Analoge inputs?

0V = 0

10V = 27648

Overshoot

> 10V-11,759V

→ 27649 - 32 511

Overflow

11,760-11,852V

→ 32515 - 32767

A.3.4 Analog inputs

Table A- 35 Analog inputs

Technical data	Description
Number of inputs	2
Type	Voltage (single-ended)
Full-scale range	0 to 10 V
Full-scale range (data word)	0 to 27648
Overshoot range	10.001 to 11.759 V
Overshoot range (data word)	27,649 to 32,511
Overflow range	11.760 to 11.852 V
Overflow range (data word)	32,512 to 32,767
Resolution	10 bits
Maximum withstand voltage	35 VDC
Smoothing	None, Weak, Medium, or Strong See the table for step response (ms) for the analog inputs of the CPU (Page 720).
Noise rejection	10, 50, or 60 Hz
Impedance	≥100 KΩ
Isolation (field side to logic)	None
Accuracy (25°C / -20 to 60°C)	3.0% / 3.5% of full-scale
Cable length (meters)	100 m, shielded twisted pair

Analoge inputs

Type signalen?

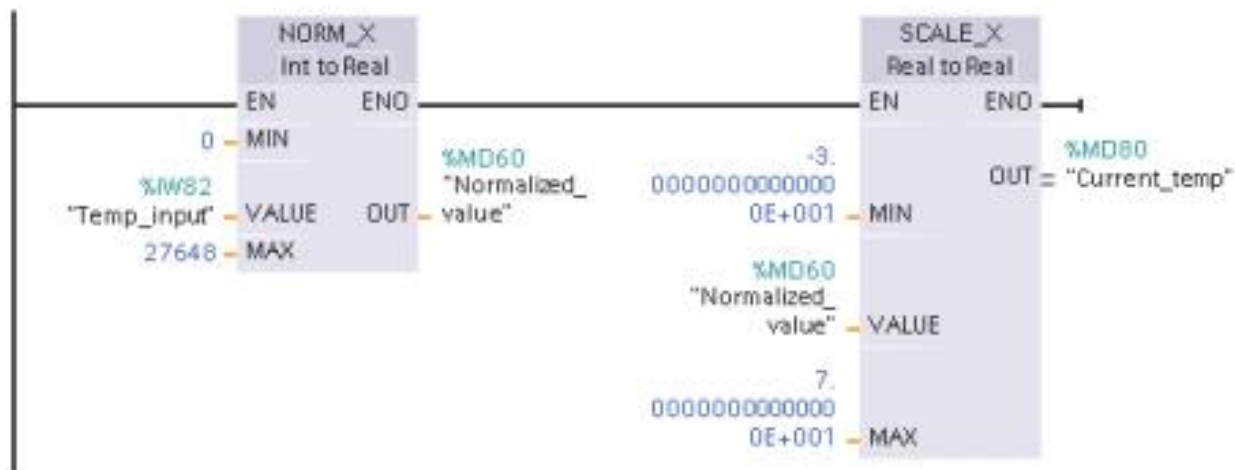
- mA vs V
- 0-20mA vs 4-20mA
- -10V – 10V vs 0-10V vs 2-10V

- Ingebouwde ingangen in de PLC: voltage!
- Andere sensoren nodig? → andere modules bijkoppelen!!!!



Analoge inputs:

Hoe programmeren?



Note that if the analog input was from an analog signal module or signal board using voltage, the **MIN** value for the **NORM_X** instruction would be **-27648** instead of 0.

Analoge outputs?

Hoe programmeren?

normalizing and scaling an analog output value

An analog output to be set in an analog signal module or signal board using output in current must be in the range 0 to 27648 for valid values. Suppose an analog output represents a temperature setting where the 0 value of the analog input represents -30.0 degrees C and 27648 represents 70.0 degrees C. To convert a temperature value in memory that is between -30.0 and 70.0 to a value for the analog output in the range 0 to 27648, you must normalize the value in engineering units to a value between 0.0 and 1.0, and then scale it to the range of the analog output, 0 to 27648:



Note that if the analog output was for an analog signal module or signal board using voltage, the MIN value for the SCALE_X instruction would be -27648 instead of 0.

Oefening Analooog? (1)

Inlezen analoge sensor?

Visualiseer op een touch screen de waarde van een analoge sensor.

Zorg er voor dat deze waarde begrijpbaar is.

Zorg er voor dat indien de waarde lager is dan een bepaald minimum dat dan de rode lamp brandt. Indien de waarde tussen twee andere waarden ligt, dan brandt de oranje lamp.

Is het boven een bepaalde waarde dan brandt de groene lamp.



Oefening Analooog? (2)

Frequentieregelaar

Een drive (VAT 20) stuurt een motor aan. (zie oefening Mr Van Den Hende)
Maak een programma voor deze oefening.

Je zorgt er voor dat de frequentieregelaar de motor aanstuurt. De snelheid (in Hz) kunnen op de HMI ingegeven worden. Vanop de HMI kun je ook de draazin instellen, net alsook start en stop.

