

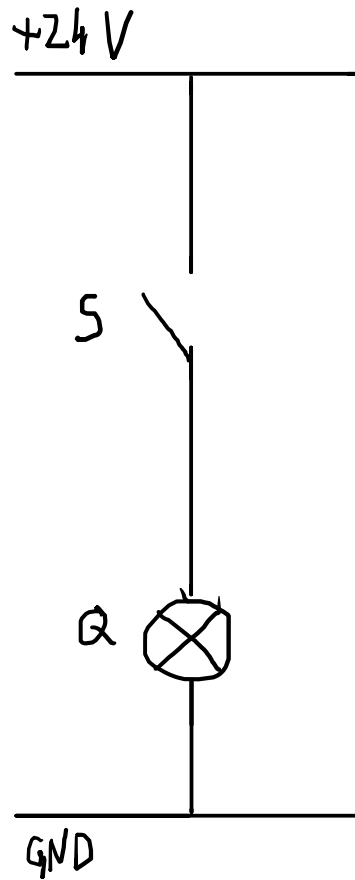
Programirljivi logični krmilnik: programiranje - 1. del

Procesna avtomatika

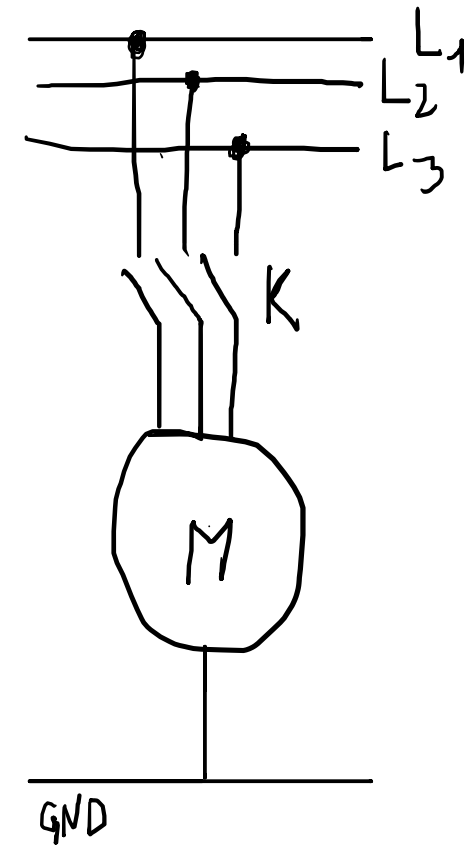
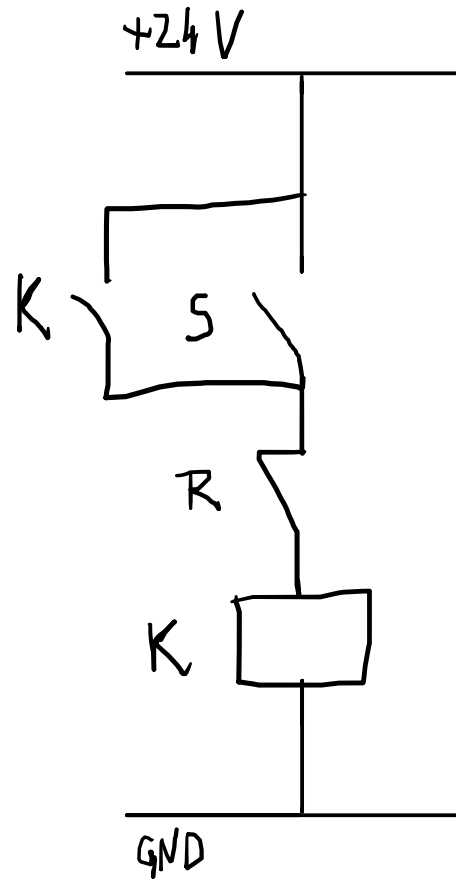
Uroš Lotrič, Nejc Ilc

Primer: programiranje brez PLK

Primer: stikalo in žarnica



Primer: krmiljenje elektromotorja



Razvoj standarda

Prej

- Lestvični diagrami slabo strukturirani
- Kompleksni visokonivojski jeziki za določene aplikacije in jeziki z namenom poenostavljanja programiranja

Standard IEC 61131

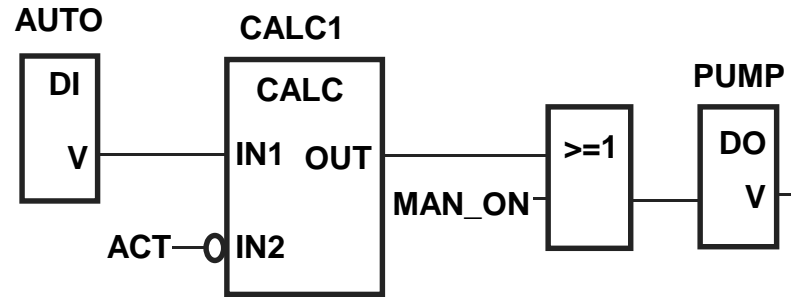
- Postavljen leta 1993
- Poskus poenotenja raznoraznih konceptov programiranja PLK
- Definira programske strukture in programske jezike:
 - lestvični diagram (ladder diagram)
 - funkcijski načrt (function block diagram)
 - lista ukazov (instruction list)
 - diagram poteka (sequential function chart)
 - strukturirano besedilo (structured text)

Zdaj

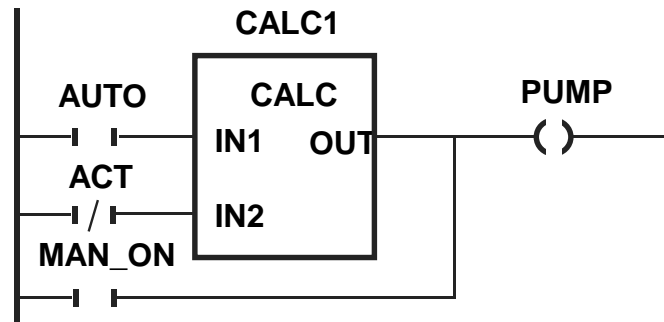
- Programi skorajda prenosljivi med različnimi proizvajalci ali serijami,
- koncepti programiranja pa zagotovo

Standard IEC 61131-3

- Funkcijski načrt (FBD)



- Lestvični diagram (LD)

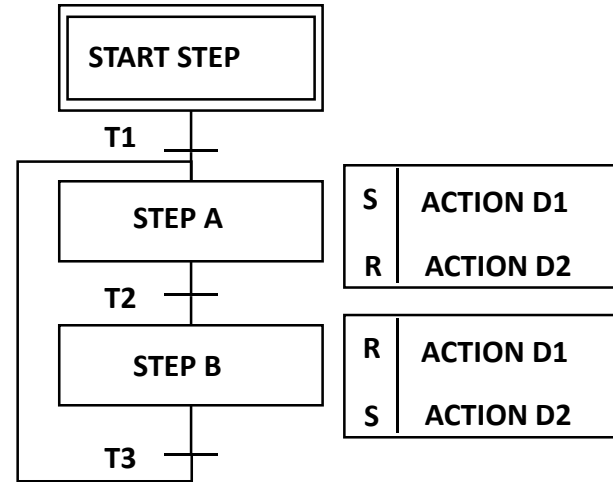


- Lista ukazov (IL)

```

A: LD    %IX1  (* PUSH BUTTON *)
   ANDN  %MX5  (* NOT INHIBIT *)
   ST    %QX2  (* FAN ON *)
  
```

- Diagram poteka (SFC)



- Strukturirano besedilo (ST)

```

VAR CONSTANT
X : REAL := 53.8; Z : REAL; END_VAR
VAR aFB, bFB : FB_type; END_VAR

bFB(A:=1, B:='OK');
Z := X - INT_TO_REAL (bFB.OUT1);
IF Z>57.0 THEN aFB(A:=0, B:="ERR");
ELSE aFB(A:=1, B:="Z is OK");
END_IF
  
```

Tipične programske strukture in podatkovni tipi

- Digitalni vhodi
- Digitalni izhodi
- Notranje binarne spremenljivke
 - Uporabniške/sistemske
 - Ohranjanje vrednosti ob izklopu (retentive)
- Časovniki
 - Ob izpolnjenem pogoju meri čas. Ko doseže nastavljeni čas sproži svoj kontakt.
- Števniki
 - Ob izpolnjenem pogoju šteje impulze.
- Združevanje binarnih spremenljivk
 - bajt (byte) – 8 bit, beseda (word) – 16 bit, dvojna beseda (double word) – 32 bit
- Celoštevilčne spremenljivke (16- ali 32-bitne)
- Necela števila v formatu plavajoče vejice (le zmogljivejši PLK)
- Nizi, čas, datum

Uporabniški bloki

Organizacijski bloki – OB

- Vmesnik med operacijskim sistemom in uporabniškim programom
- Vsak ima svoj pomen in svojo nespremenljivo identifikacijsko številko

Funkcije – FC

- Pogosto ponavljajoči deli programa (razhroščevanje!)
- Lastnosti jim določimo s klicnimi parametri
- Lahko vrnejo funkcijsko vrednost in tudi dodatne izhodne parametre

Podatkovni bloki – DB

- Vsebuje podatke o uporabniškem programu
- Obliko jim lahko določimo sami
- Globalni podatkovni bloki in podatkovni bloki, ki pripadajo funkcijskim blokom

Funkcijski bloki – FB

- Del uporabniškega programa
- Statično lahko hranijo lokalne spremenljivke v dodeljenem podatkovnem bloku
- Funkcijski blok v programu lahko kličemo večkrat, vsaka instanca ima lahko svoj podatkovni blok (razhroščevanje!)

Uporabniški bloki

Vsak blok ima

- Glavo
 - ime bloka, avtor, verzija, zaščita pred pregledovanjem (ang. know-how protection)
- Deklaracije
 - Začasni lokalni podatki za shranjevanje vmesnih rezultatov (OB, FC, FB) – sklad L
 - Parametri (FC, FB)
 - Vhodi, izhodi, vhodi/izhodi
 - Statični lokalni podatki (FB)
- Programski del ali inicializacija
 - Ukazi (OB, FC, FB) ali
 - Nastavitev začetnih vrednosti spremenljivk (DB)

Klicanje bloka

- Pri FC je potrebno navesti vse parametre
- FB za parametre, ki pri klicu niso navedeni, uporabi privzete vrednosti

Organizacijski bloki

Program na PLK je sestavljen iz

- Operacijskega sistema
- Uporabniškega programa

Organizacijski bloki

- So del uporabniškega programa
- Kliče jih operacijski sistem
- Razvrščeni so v prioritete razrede
 - Višja prioriteta se izvede prej
- število OB je nespremenljivo in odvisno od modela

Organizacijski blok	Klic	Prioriteta
OB1	Ciklično klicanje	1
Časovni dogodki OB10 .. OB17	Ob določenem času	2
Časovne zakasnitve OB20 .. OB23	Klic po preteku časa	3 .. 6
Ciklične prekinitve OB30 .. OB38	Ob podanih časovnih intervalih	7 .. 15
Strojne prekinitve OB40 .. OB47	Prekinitveni signali V/I	16 .. 23
Asinhrona napake OB80 .. OB87	Napake strojne opreme	26 (28)
Zagon OB100 .. OB102	Ob zagonu krmilnika	27
Sinhrona napake OB121 .. OB122	Napake v uporabniškem programu	Izvajani OB

Procesiranje

Zagonski program

Glavni program

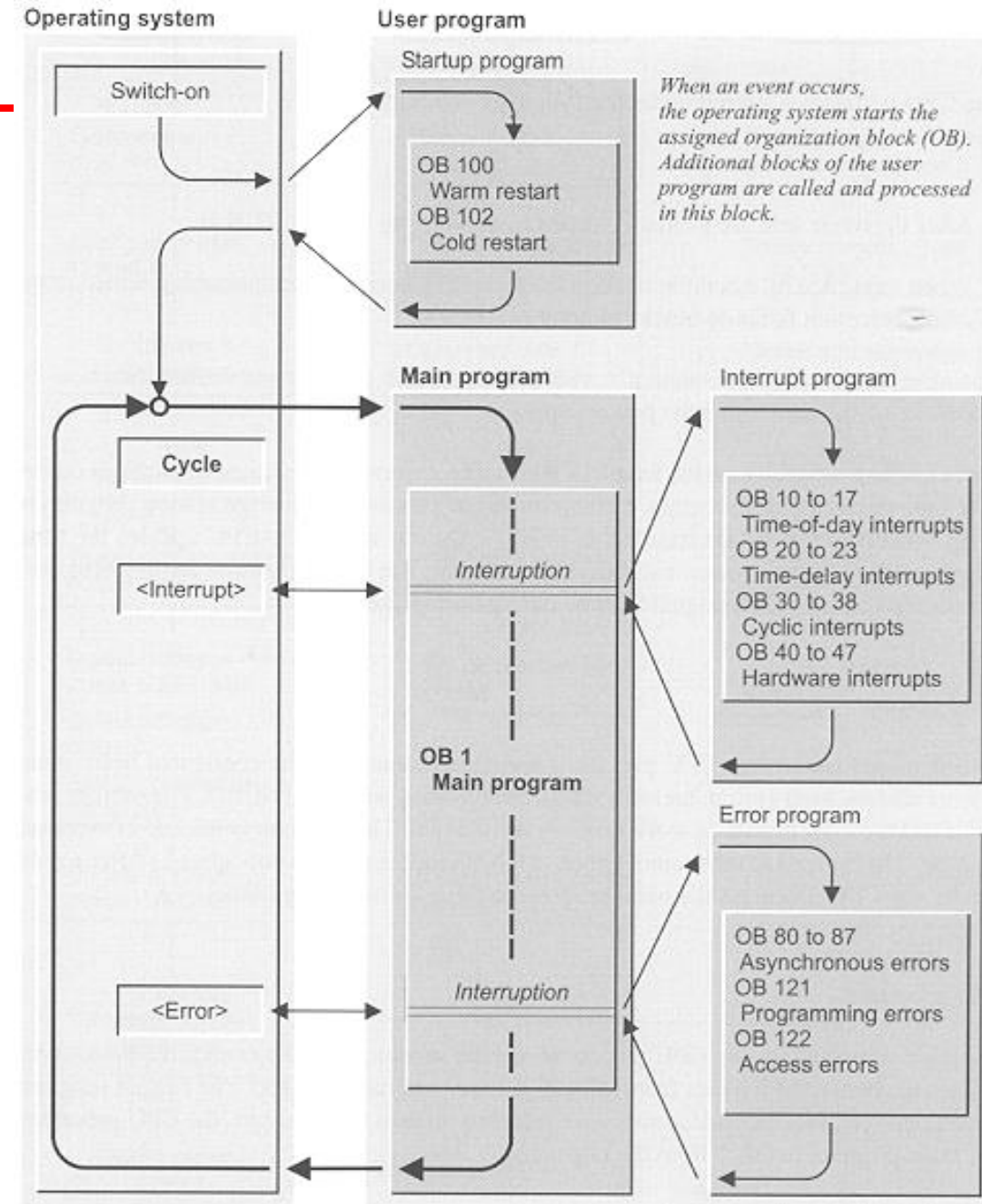
- Ciklično klicanje OB1
- OB1 ima najnižjo prioriteto

Prekinitveni program

- Izvede se, če ima višjo prioriteto kot trenutno izvajana koda

Ukrepanje ob napakah

- Asinhrona napake
 - Izguba napajanja na razširitveni enoti
- Sinhrona napake
 - Naslavljanje neobstoječega elementa
 - Napaka pri pretvarjanju podatkov



Zagon

Zagonski program se izvede

- Ko procesor dobi napajalno napetost
- Pri preklopu iz načina STOP v način RUN
- Če to zahteva druga naprava (komunikacija)

Hladen zagon

- Zbrišejo se vsi podatki v sistemskem pomnilniku (slika vhodov, izhodov, spominski biti, časovniki, števniki, lokalne spremenljivke), tudi tisti, ki naj bi se ohranjali (ang. retentive)
- Zbrišejo se uporabniški program in podatki

Topel zagon

- Zbriše samo podatke iz sistema pomnilnika, ki naj se ne bi ohranjali
- Uporabniški program in podatki se ohranijo

Vroč zagon

- Program se nadaljuje tam, kjer se je končal

Resetiranje in ohranjanje pomnilnika

Reset pomnilnika postavi procesor v osnovno stanje

- Način STOP → Način MRES za 9 s (lučka stop neprekinjeno gori) → način STOP za manj kot 3 s → način MRES za kratek čas
- Ohranijo se
 - Nastavitve protokola MPI
 - Ura realnega časa (RTC)
 - Diagnostični pomnilnik

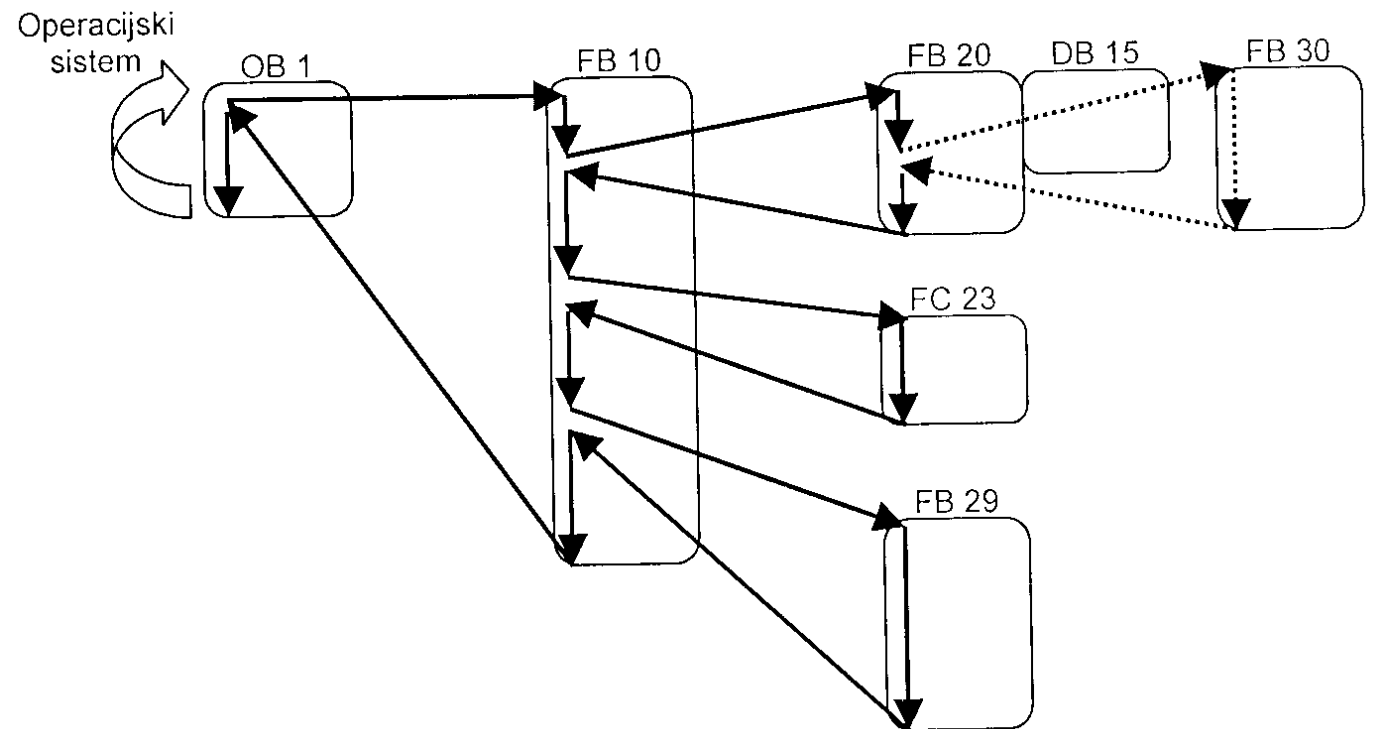
Ohranjanje vrednosti

- Del pomnilnika se ohranja (je retentiven) pri toplém zagonu. Ohranjajo se lahko
 - Spominski biti
 - Časovniki
 - Števniki
 - Vsi podatkovni bloki

Glavni program

Ciklično izvajanje uporabniški program (OB1)

- Običajno za PLK
- Dogodkovno procesiranje - dodatek
- Izvaja se samo v načinu RUN
- Gnezdenje klicev omejeno z velikostjo blokovnega sklada (B)



Prekinitve

Ob prekinitvi se kliče ustrezni blok OB

Strojne prekinitve

- Opazovanje in hitro reagiranje na kritične dogodke v procesu
- Ustrezen V/I modul, ki mu je potrebno omogočiti klice prekinitev
- Spreminjanje, onemogočanje med delovanjem s sistemskimi funkcijami

Ciklične prekinitve

- Proženje ob natančno določenih intervalih od 1 ms do 1 min

Časovni dogodki

- Ko naj se izbrana funkcija izvede samo v točno določenem času
- Enkratno ali periodičen dogodek (min, ura, dan, teden, mesec, leto)

Časovne zakasnitve

- Primerljivo s časovniki
- Neodvisno od časovnikov, bolj natančno

Lovljenje napak

Vzroki napak

- Napaka v uporabniškem programu
 - Aritmetične operacije (preliv, deljenje z 0, ...)
 - Naslavljanja
- Napaka v strojni opremi
 - Predolgo izvajanje cikla
 - Baterijsko napajanje PLK
 - Diagnostična prekinitev
 - Sprememba strojne konfiguracije PLK-ja
 - Težave s komunikacijo
 - Klic neobstoječega organizacijskega bloka
 - Težave s stojno opremo

Za vsako napako obstaja svoj organizacijski blok

- Če ustrezen organizacijski blok ne obstaja, procesor pade v način STOP

Časovniki in števniki

Proženje časovnika/števnika je vedno ob spremembi signala

Časovniki

- Namenjeni so za izvedbo časovno pogojenih postopkov
 - Začetek po določenem času
 - Določitev časa opazovanja
 - Merjenje časa med opazovanimi dogodki
 - Generiranje pulzov
- Siemens S7-315-2 PN/DP
 - 5 različnih časovnih funkcij
 - 256 časovnikov, 10 ms .. 9990 ms
 - Časovne baze: 10 ms, 100 ms, 1 s, 10 s

Števniki

- Namenjeni so štetju dogodkov
- Sprememba stanja ob spremembi vhoda
- Čas med dvema dogodkoma mora biti daljši od programskega cikla
- Siemens S7-315-2 PN/DP
 - 256 števnikov
 - Štejejo od 0 .. 999

Naslavljanje

Globalna naslovna področja

- Vhodi perifernih naprav (PI)
- Izhodi perifernih naprav (PQ)
- Vhodi (I)
- Izhodi (Q)
- Spominski biti (M)

Naslavljanje globalnih podatkovnih blokov (DB)

- Pred naslavljanjem s kratkim naslovom ga je potrebno odpreti
- Pri naslavljanju s polnim naslovom se odpre avtomatsko

Naslavljanje začasnih lokalnih spremenljivk (sklad)

- L...

Naslavljanje statičnih lokalnih spremenljivk

- DI...

Naslavljanje

Naslovno področje	Oznaka	Bit	Byte	Word	Double Word
Periferni vhodi	PI	-	PIB y	PIW y	PID y
Periferni izhodi	PQ	-	PQB y	PQW y	PQD y
Vhodi	I	I y.x	IB y	IW y	ID y
Izhodi	Q	Q y.x	QB y	QW y	QD y
Spominski biti	M	M y.x	MB y	MW y	MD y
Globalni podatkovni bloki	DB	DBX y.x DB z.DBX y.x	DBB y DB z.DBB y	DBW y DB z.DBW y	DBD y DB z.DBD y
Začasne lokalne spremenljivke	L	L y.x	LB y	LW y	LD y
Statične lokalne spremenljivke	DI	DI y.x	DIB y	DIW y	DID y
Časovniki	T	T n			
Števniki	C	C n			

z – številka DB
n – številka časovnika ali števca

y – številka bajta
x – številka bita v bajtu

Naslavljanje

Primeri

- I 1.0 – vhod: bit 0 v bajtu 1 (1 bit)
- QB 2 – izhodni bajt 2 (8 bitov)
- IW 4 – vhodni word 4 (16 bitov)
- QD 24 – izhodni double word (32 bitov)
- DB 10.DBX 2.0 – bit 0 v bajtu 2 v podatkovnem bloku 10

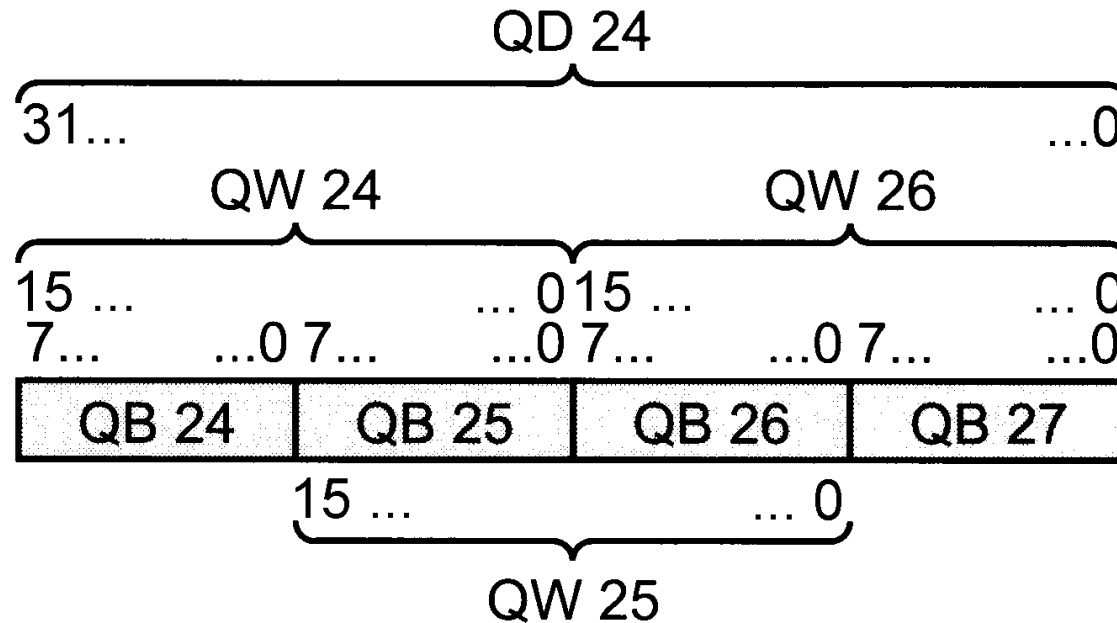
Naslavljanje s simboli

- DB 10 – “Motor”
- bit 2.0 – “Zavora”
- DB 10.DBX 2.0 – “Motor”.Zavora

Naslavljanje

Bit, Byte, Word in Double Word v pomnilniku

- Pravilo debelega konca: naslov operanda je enak naslovu besede, ki vsebuje najpomembnejši (debeli) del operanda



Naslavljanje: vnos konstant

Podatkovni tip	Primer
BOOL	FALSE .. TRUE
BYTE	16#00 .. 16#FF
WORD	16#0000 .. 16#FFFF 2#0000_0000_0000_0000 C#999
DOUBLE WORD	16#0000_0000 .. 16#FFFF_FFFF 2#0000_0000_..._0000_0000 B#(255,255,255,255)
TIME za časovnik	T#0ms .. T#2h46m30s
INT	-32768 .. 32767
DINT	L#-2147483648 .. L#2147483647

Lestvični diagrami: standard

Najbolj uporabljan jezik za PLK

Grafični jezik, ki omogoča enostaven prehod iz električnih shem v programiranje

Standard 61131-3

- Navpični Tirnici
 - Osnova lestvičnega diagrama
 - Leva tirnica je priključena na visoko napetost in predstavlja logično 1
 - Desna tirnica je priključena na nizko napetost in predstavlja logično 0
- Priključeni elementi so predstavljeni z vodoravnimi povezavami
 - Stanje priključenega elementa je definirano kot “vklopljen” ali “izklopljen” in ustreza logičnim vrednostim 1 in 0. V prvem primeru prepušča električni tok, v drugem ne.
- Vodoravna povezava
 - Stanje vodoravne se spreminja glede na to ali skozi element teče električni tok ali ne
 - Kadar je element vklopljen vodoravna povezava prenaša stanje vodoravne povezave na levi strani elementa na vodoravno povezavo na njegovi desni strani
- Navpične povezave povezujejo enega ali več priključenih elementov
 - Stanje navpične povezave je “izklopljeno”, če so vse vodoravne povezave izklopljene
 - Stanje navpične povezave je “vklopljeno”, če je vsaj ena vodoravna povezava vklopljena
 - Stanje navpične povezave se prenese na vse vodoravne povezave na njeni desni strani
 - Prenos stanja navpične povezave na levo stran ni dovoljeno

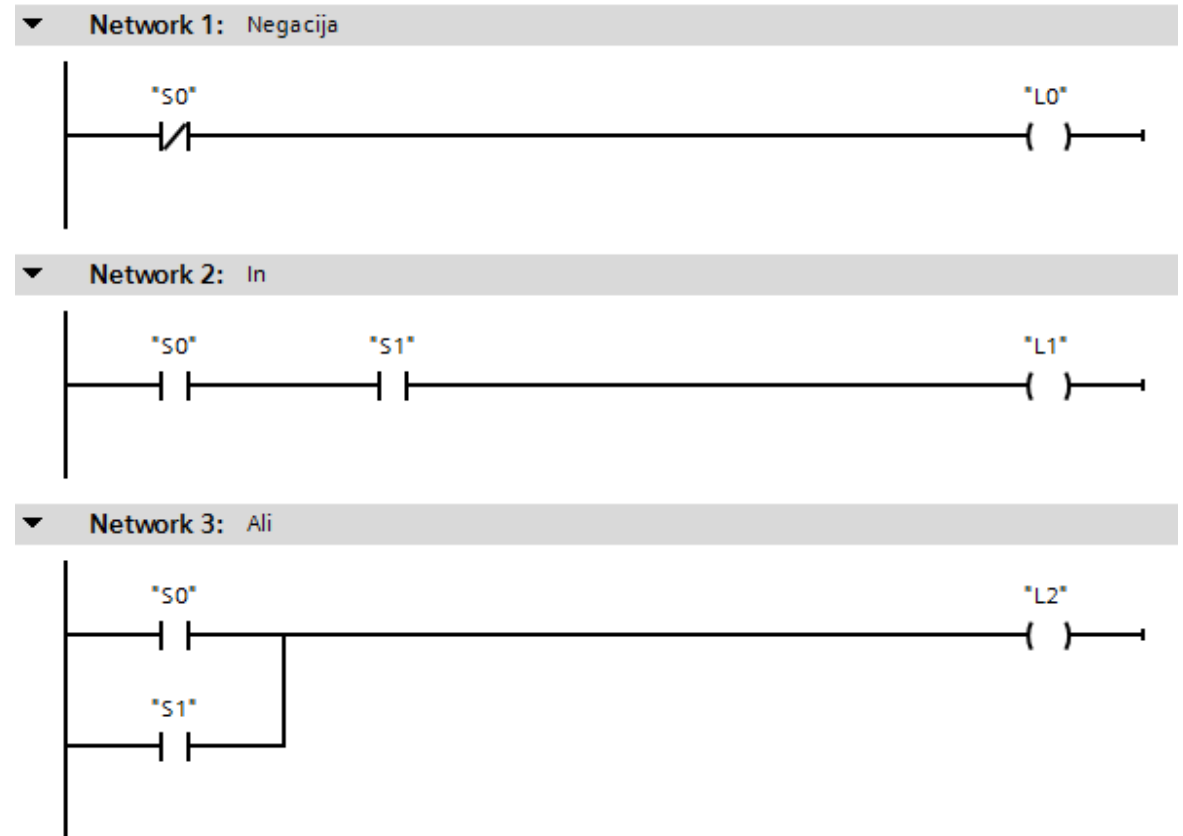
Lestvični diagrami: standard

Standard 61131-3

- Elementi in povezave tvorijo na lestvi kline (Siemens: Network)
- Običajno ima vsak klin množico ustrezno povezanih vhodnih pogojev s katerimi nadziramo eno ali več tuljav (izhodov)
 - Dobra praksa: vsaka tuljava (izhod) naj se v programu pojavi samo na enem mestu

Primer

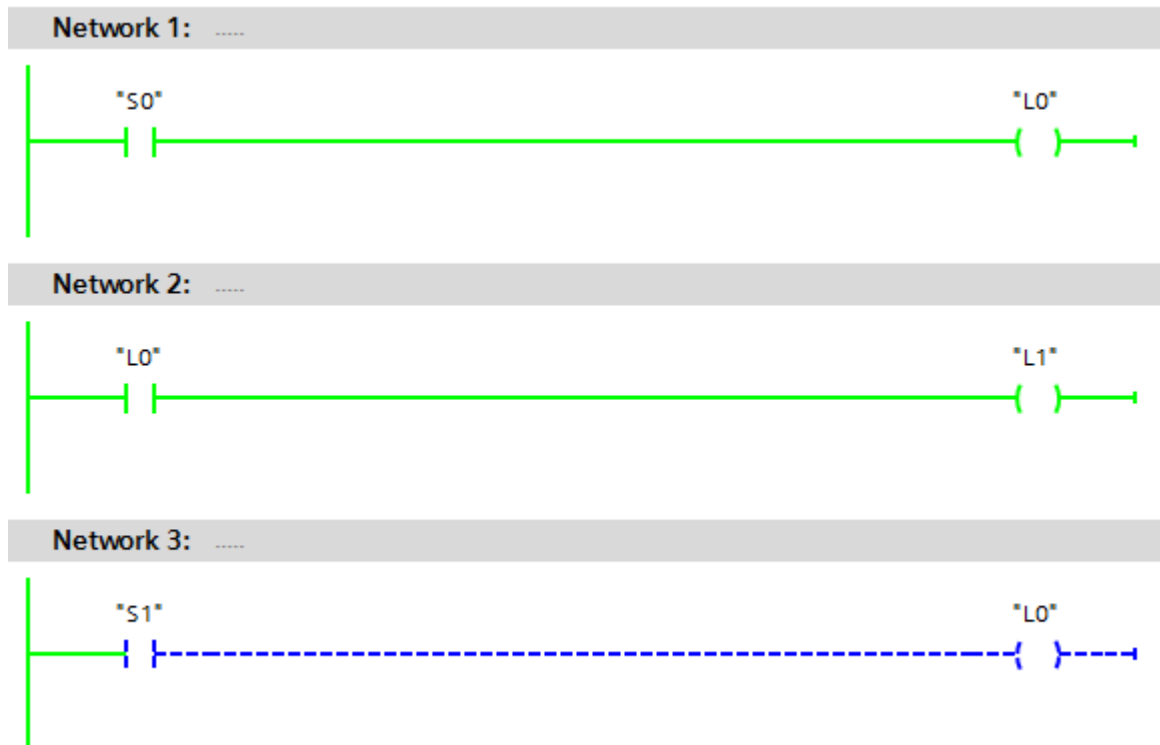
- Osnovne logične operacije



Lestvični diagrami: spremembe

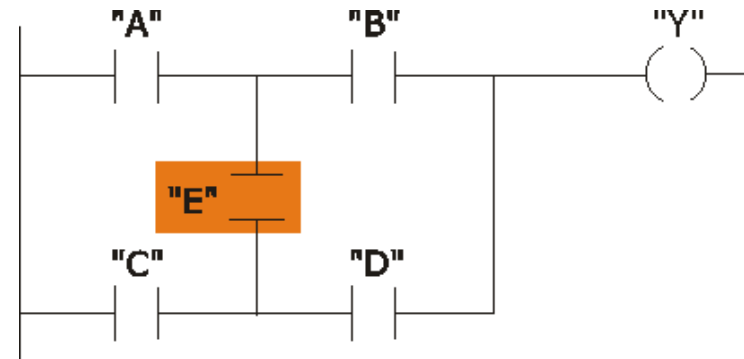
Spreminjanje vhodov in izhodov

- Vhodi so skozi cel programski cikel konstantni (paralelno)
- Izhodi se spreminjajo klin za klinom (sekvenčno)
 - Na fizične izhode se prenese zadnja vrednost izhoda v pomnilniški sliki



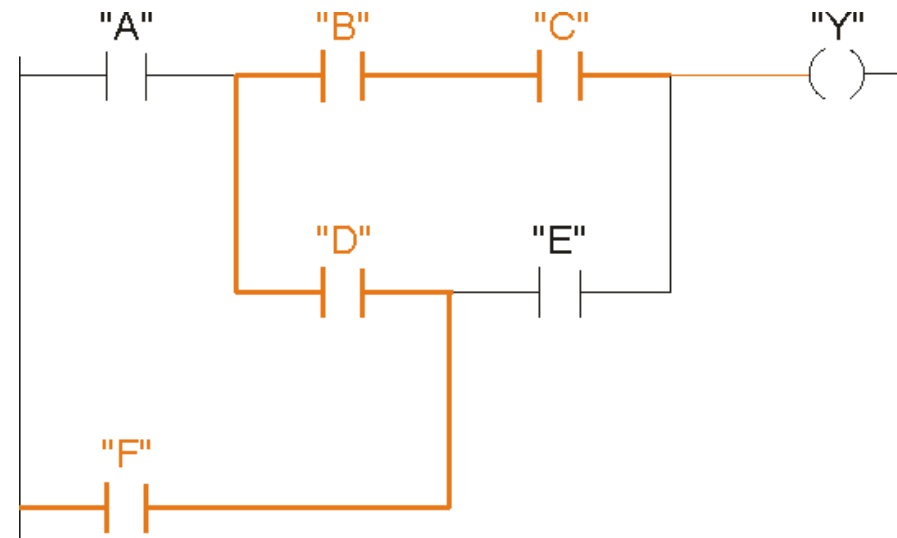
Lestvični diagrami: omejitve

Navpičnih vezav elementov ni



Tok teče samo iz leve na desno

- Običajno orodja izdelave takih vezij ne omogočajo



Lestvični diagrami: razširitve

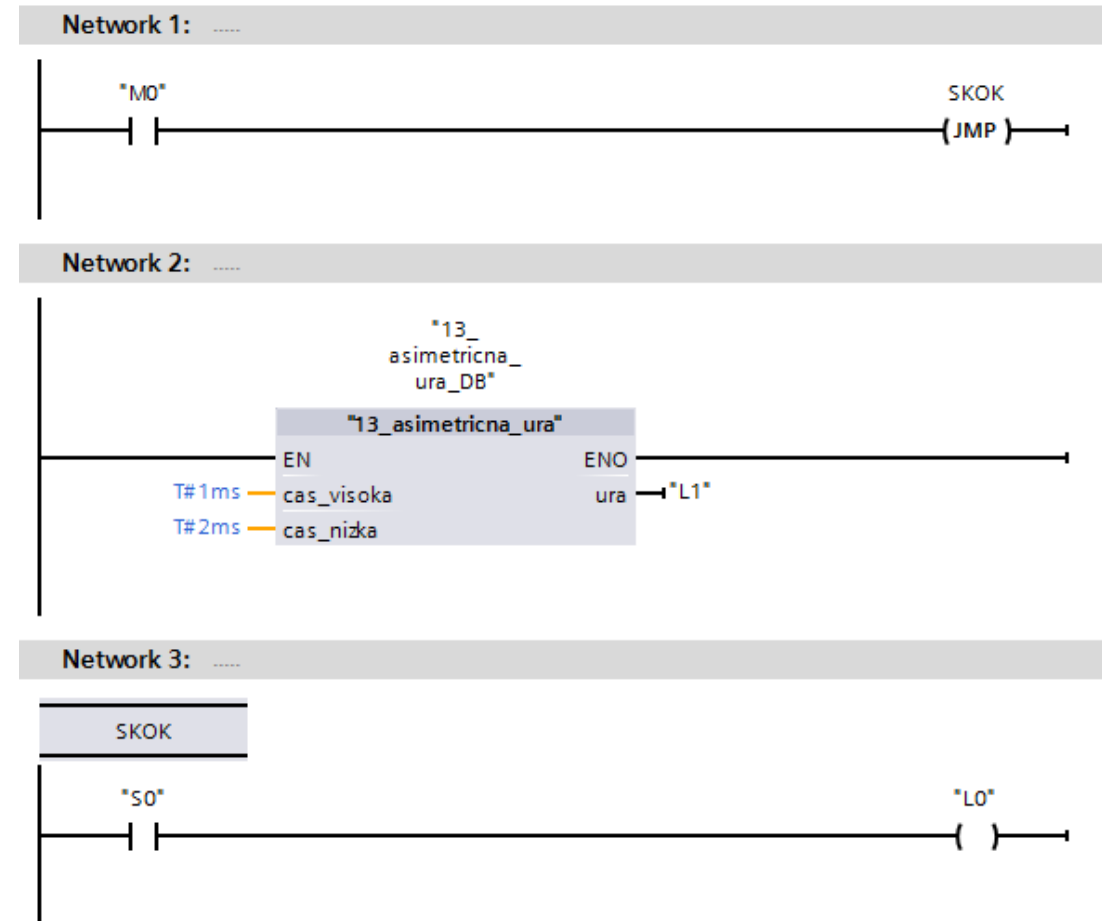
Osnovni lestvični diagram pozna samo stikala in releje

Nujne razširitve

- Klici funkcij in podprogramov
- Skoki
- Uporaba strukturiranih podatkovnih tipov
- Procesiranje analognih vrednosti

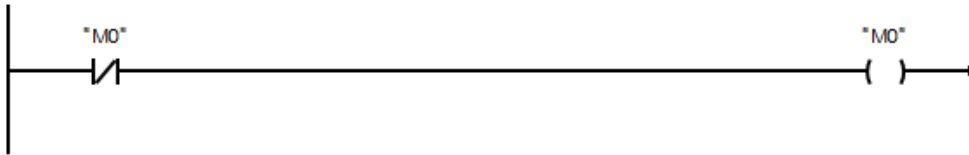
Programi postanejo težje berljivi

- Intuitivnost stikal in relejev se izgubi
- Niso primerni za velike kompleksne projekte

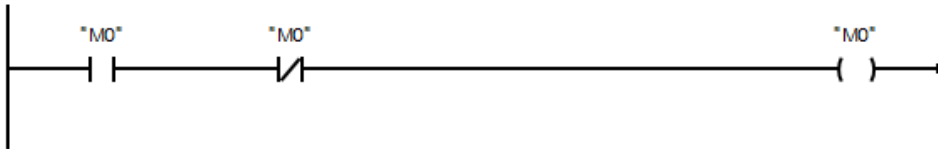


Lestvični diagrami: primeri

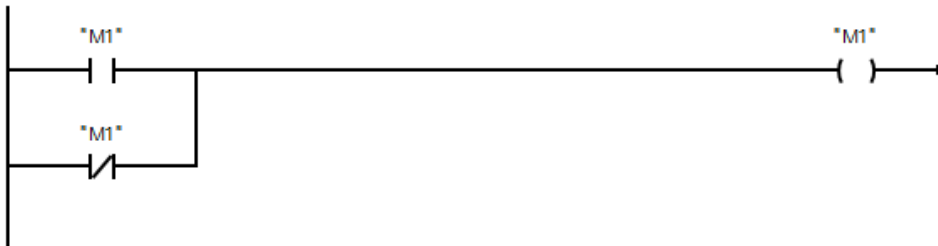
Network 1: Ura (sprememba vsak cikel OB1)



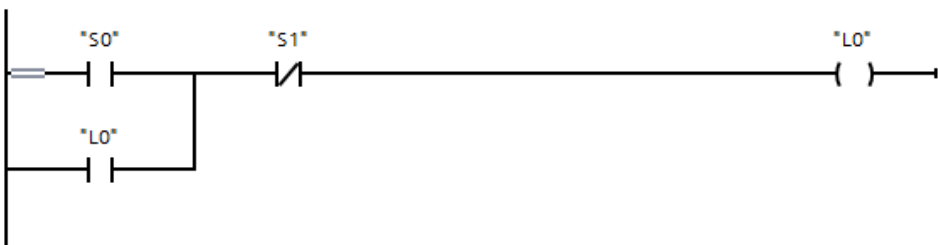
Network 1: False (izhod bi lahko bil poljuben, samo tako preprečimo, da bi ga kdo preglasil od zunaj)



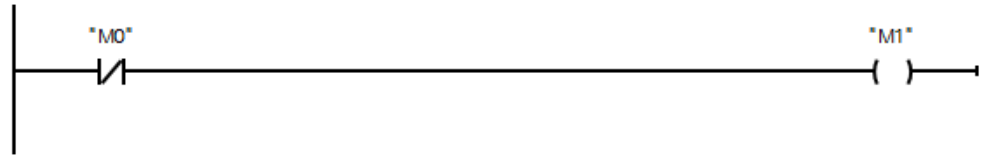
Network 2: True



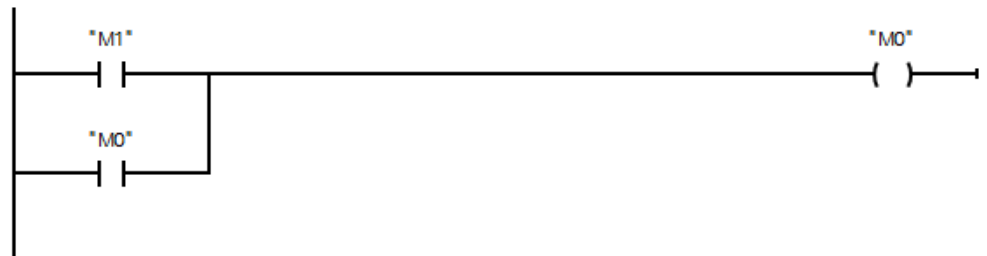
Network 1: Primer: krmiljenje elektromotorja - pomnilna celica (S0=set, S1=reset, L0=Q)



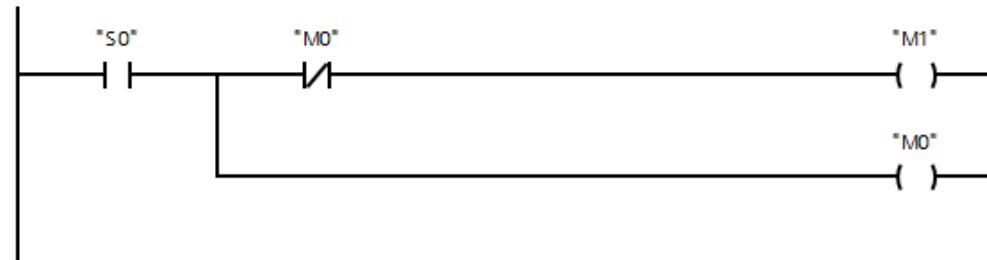
Network 1: M1 bo True samo prvi cikel izvajanja tega bloka



Network 2:



Network 1: Detekcija pozitivne fronte na S0



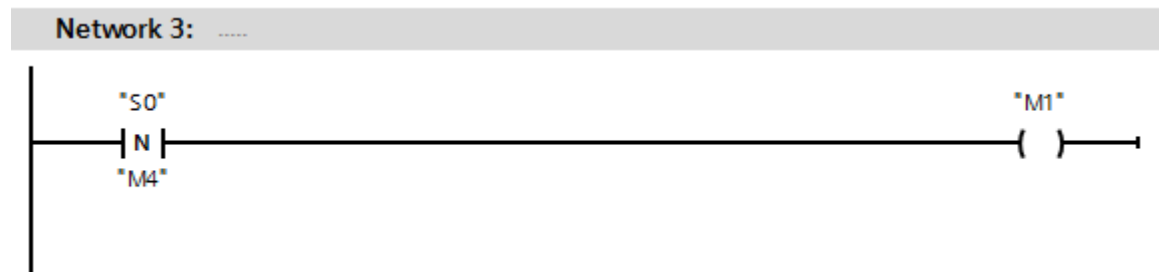
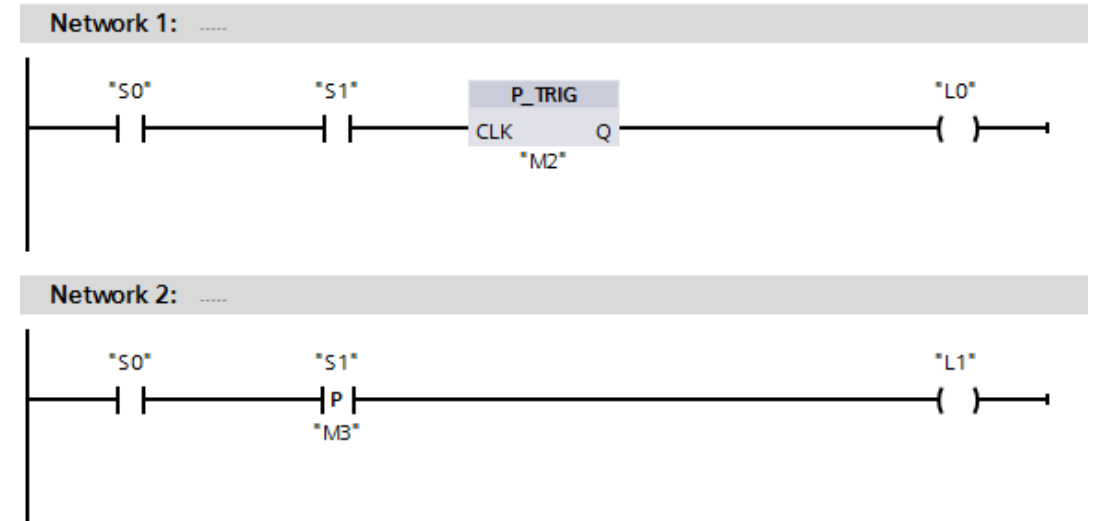
Lestvični diagrami: zaznavanje fronte

Zaznavanje pozitivne fronte

- P_TRIG zaznava pozitivno fronto celotnega izraza pred njim

Zaznavanje negativne fronte

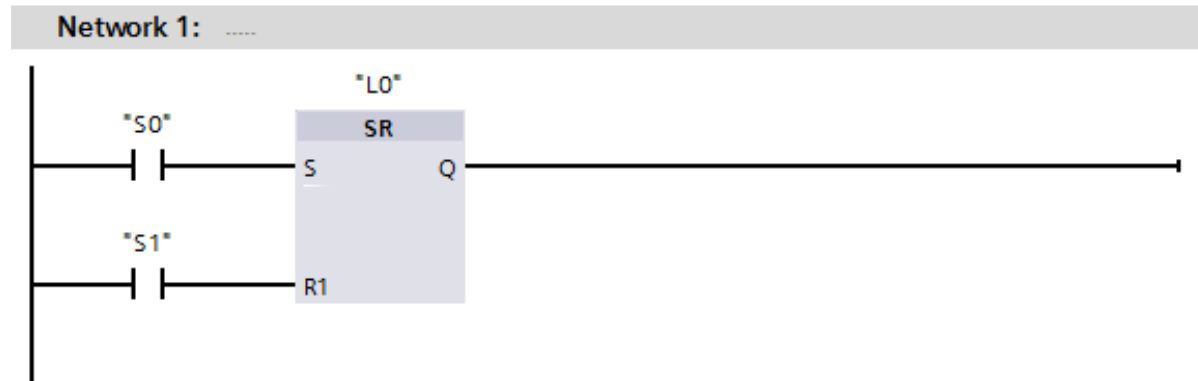
- bit M1 je aktiven en cikel



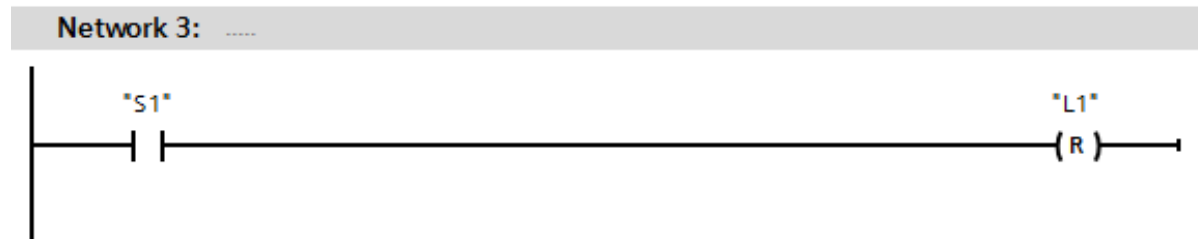
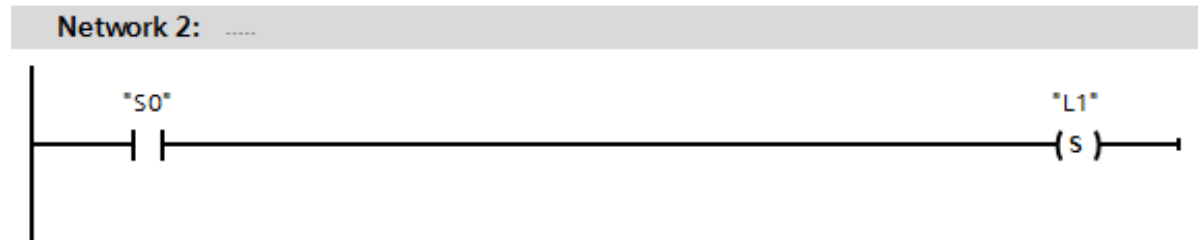
Lestvični diagrami: pomnilna celica

Pomnilna celica SR (reset ima prednost pred set)

- Način z blokom

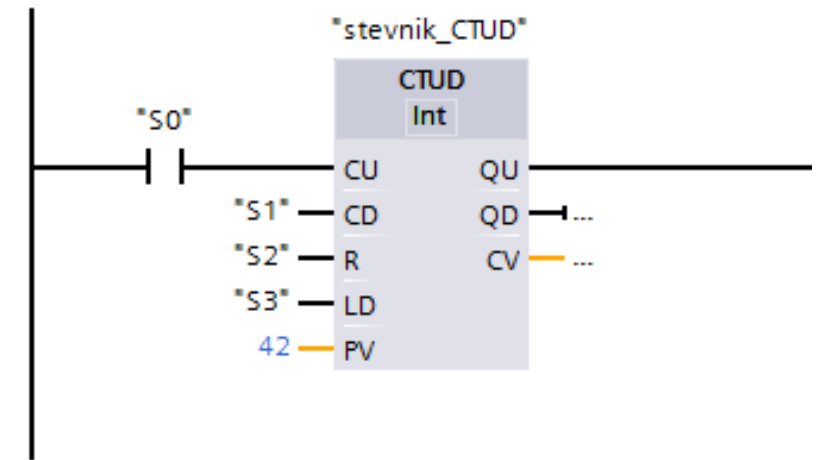
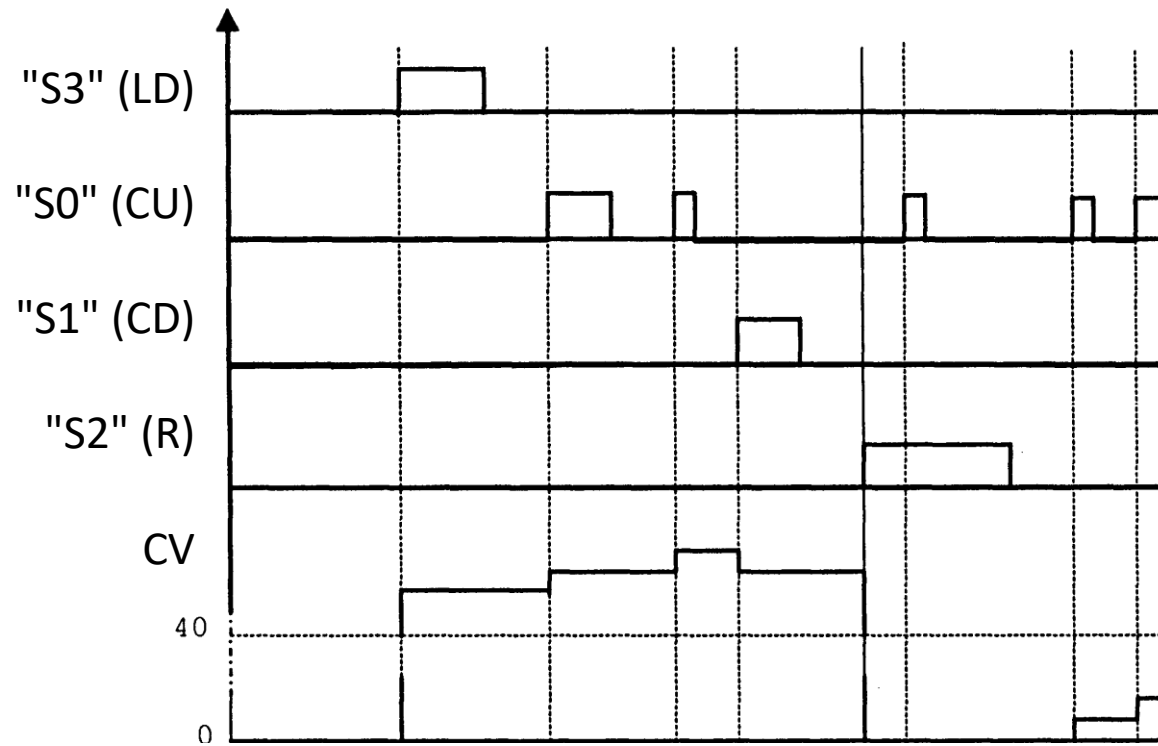


- Način s tuljavami



Lestviční diagrami: števniki

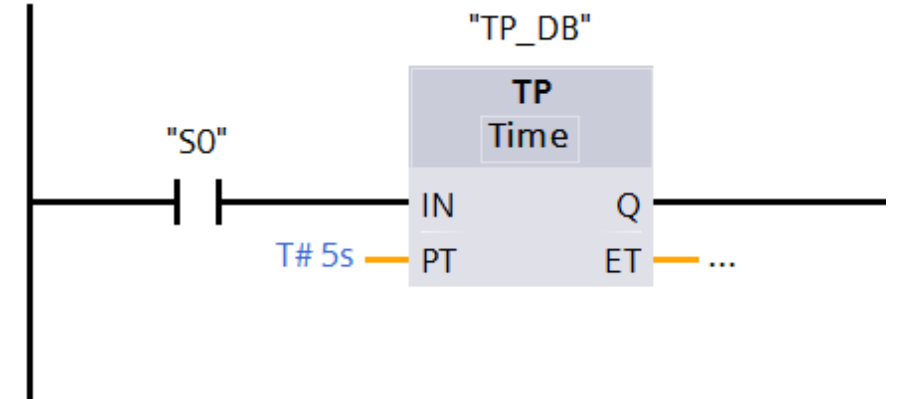
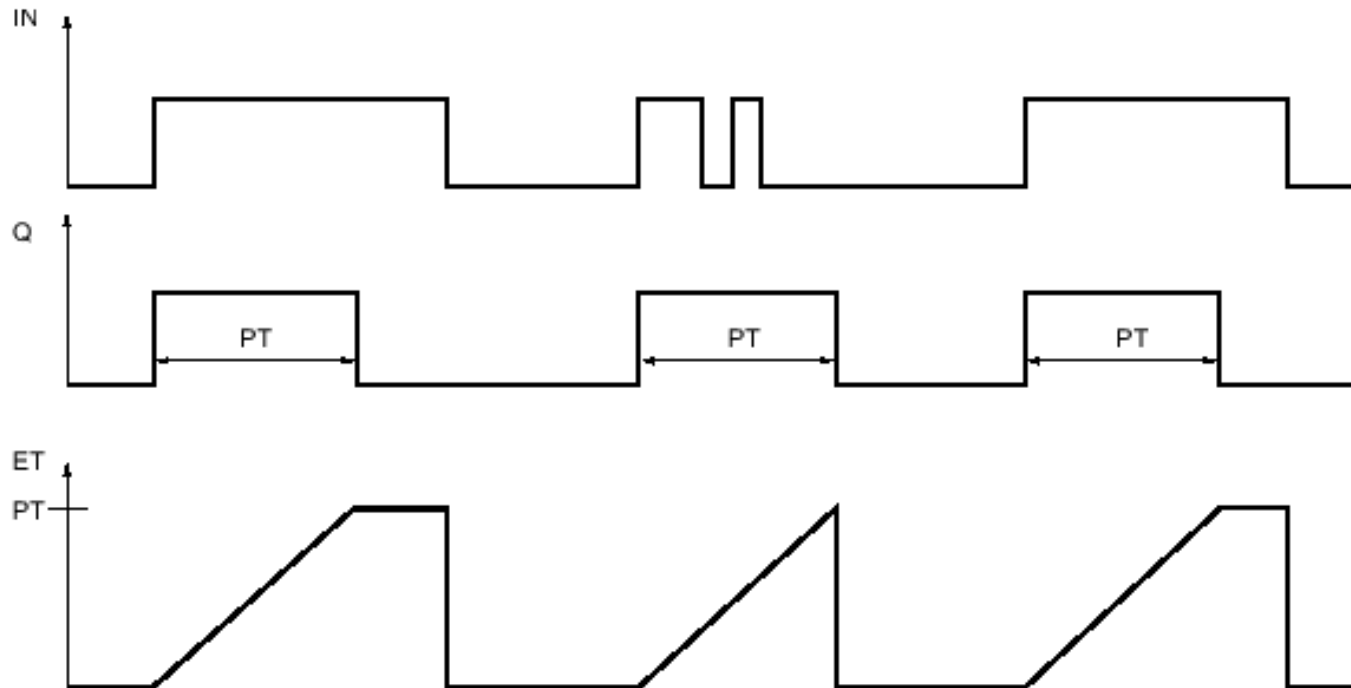
Časovni potek



Lestviční diagrami: časovníki

Pulz (timer pulse)

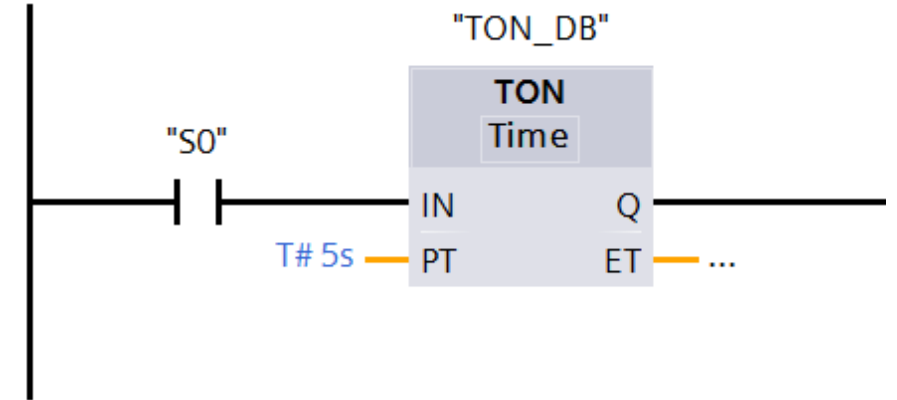
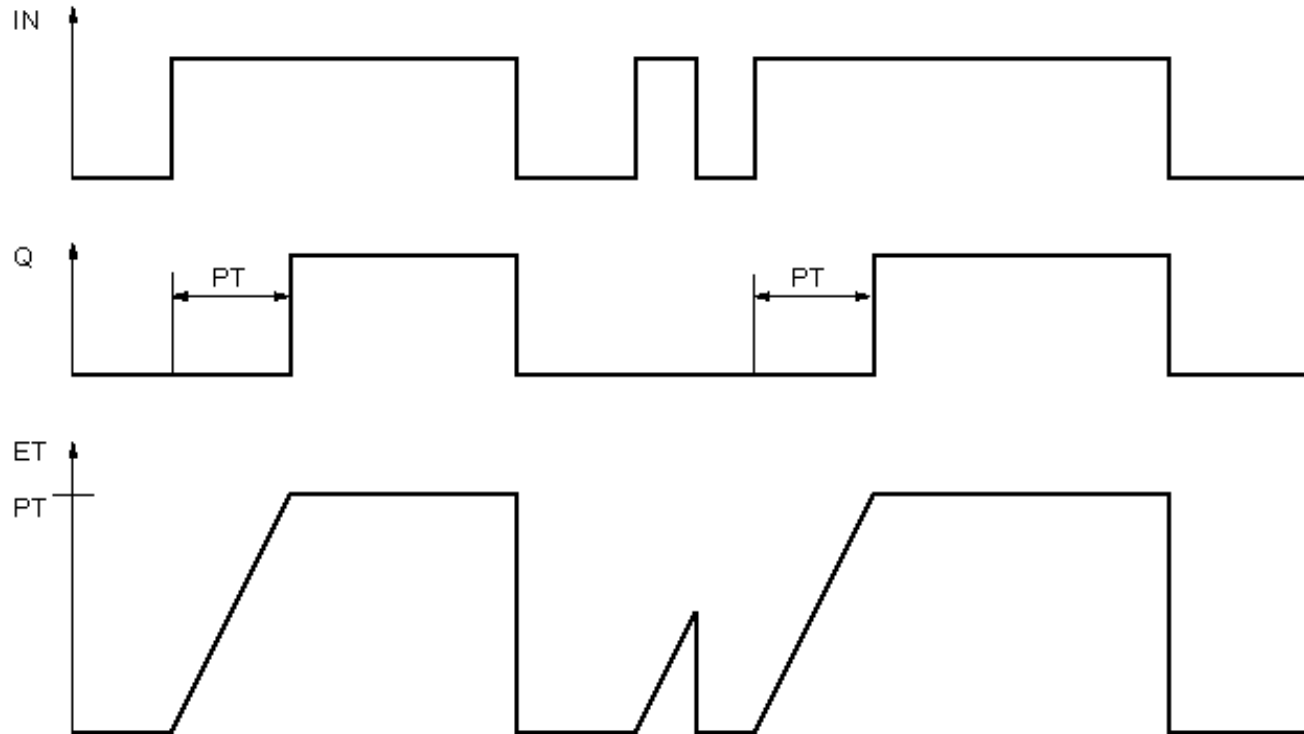
- Časovni potek



Lestviční diagrami: časovníki

Zakasnitev vklopa (timer on-delay)

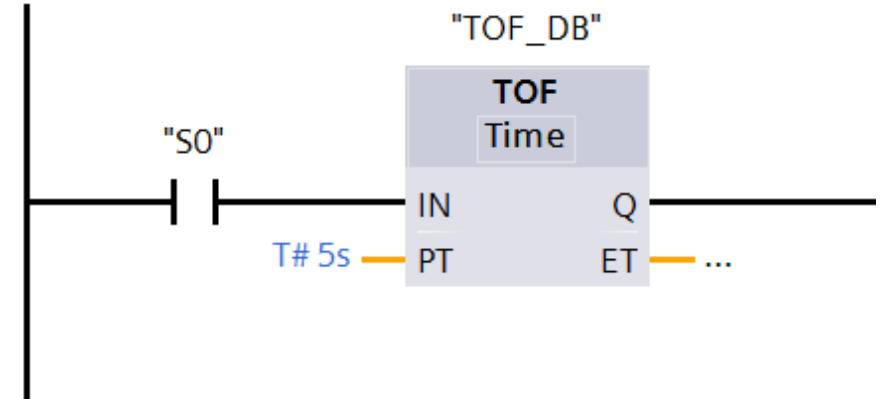
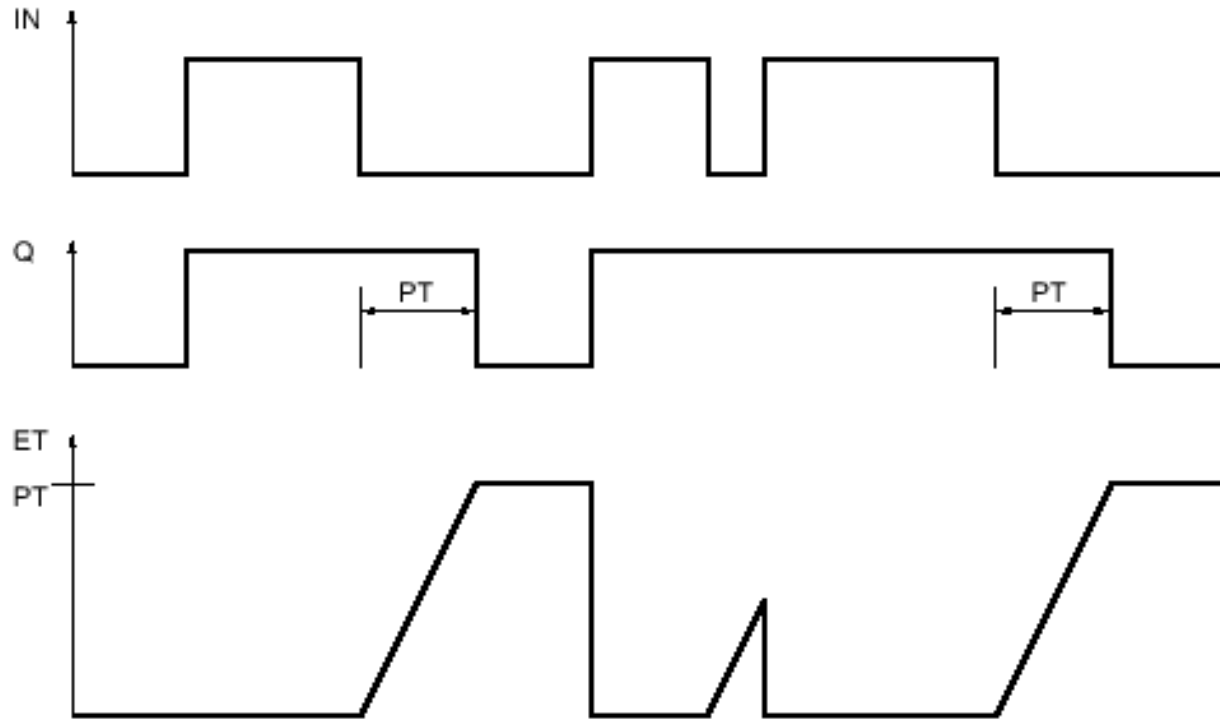
- Časovni potek



Lestvični diagrami: časovniki

Zakasnitev izklopa (timer off-delay)

- Časovni potek



Lestvični diagrami: časovniki

Primer: blokada prehitre menjave smeri vrtenja motorja

