

**SIEMENS**

**SIMATIC**

**100 U mini PLC**

**med programmeringsenhed**

**PG 605 U**

**ABC om programmering**  
**2. udgave**

# SIEMENS

## SIMATIC 100U mini PLC og PG 605U programmeringsenhed



SIMATIC 100U mini-PLC  
og PG 605U programmeringsenhed

<u>Indholdsfortegnelse</u>	Side
<u>1. Indledning</u>	1.3
<u>2. Opstilling af udstyret</u>	2.1
Tilslutning af PLC	2.1
Tilslutning af programmeringsenhed	2.5
Simulatorenhed	2.5

<u>3. Programmering m.v.</u>	3.1
Opstart af programmering	3.2
Sletning af lager	3.3
Indlæsning	3.4
Serieforbindelse	3.5
Programoverførsel	3.6
Stop af PLC fra programmeringsenhed	3.7
Udlæsning af program	3.8
Parallelforbindelse	3.9
Programoverførsel	3.10
Start af PLC fra programmeringsenhed	3.11
Programændringer	3.12
Start-Stop (hukommelsesfunktion)	3.16
Merkere	3.19
Tidsled	3.20
Taktgenerator	3.21
Udlæsning af status	3.21
Tællere	3.22
Ændring af timere og tælleres værdier under drift	3.24
Den nemme metode	3.24
Den sikre metode	3.26
Programmering af datablok	3.25
Programkopiering	3.29
<u>4. Opgaver</u>	4.1
Øvrige danske håndbøger	4.5
Programdokumentation	4.5
<u>5. Programmeringssproget STEP-5</u>	5.1
Opbygning af en instruktion	5.2
Betydning af operandart	5.4
Oversigt over operander	5.5
Operationsoversigt STEP 5. Grundoperationer	5.6
Operationsoversigt STEP 5. Ekstra operationer	5.10
SIMATIC 100U: Tekniske data	5.12

# 1. Indledning

Denne "kom i gang vejledning" er skrevet for førstegangsbrugere af Simatic 100U mini-PLC og PG 605U programmeringsenhed.

Vejledningen her er ment som en hjælp til at komme ind på det rette spor, når 100U og 605U skal arbejde sammen.

Endvidere gives en kort beskrivelse af programmeringssproget STEP 5.

Simatic 100U er et industristyringsprodukt, udviklet på basis af over 15 års erfaring i produktion af PLC'er, med alt hvad dette indebærer af viden om styretekniske behov og barske miljøer m.h.t. transienter m.m.

Et genialt byggeklodsprincip

Den nye SIMATIC 100U er konstrueret til dit behov.

Du køber nemlig kun, hvad du virkelig har brug for - uanset om anlægget skal have 8 eller 256 tilslutningspunkter.

Selv ved køb af det mindst tænkelige system til 8 punkter, til en pris der svarer til 5 fortrådede hjælperelæer, kommer du i besiddelse af den mest fremtidssikre PLC-teknologi i Europa.

Og periferien?

Den modulære opbygning stiller dig helt frit, når du skal projektere og bestykke et anlæg. Her er alt, hvad du behøver:

- \* Digitale ind- og udgangsmoduler.
- \* Analoge indgange til strøm, spænding og modstand.
- \* Analoge udgange til strøm og spænding.
- \* Specielle moduler til eksterne timere, tællere og analog grænseværdiovervågning.

Og, det siger sig selv, at med en så fleksibel styringsenhed kan alt kombineres nøjagtigt efter dit ønske.

Hvis 100U skulle være for lille til at løse en given opgave, vil SIMATIC S5-115U være løsningen.

Den programmeres i samme sprog og med det samme programmeringsudstyr og kan udbygges til 1024 ind-/1024 udgange med 128 timere og 128 tællere. Desuden leveres ekstraudstyr til 115U, som farvemonitorsystem, bussystemer til lokalnet som f.eks. Sinec L1 og Sinec H1 (ethernet) og meget mere. Kort sagt - med SIMATIC 100U og 115U har vi markedets mest fleksible PLC-systemer.

Den tid er forlængst forbi, da du var nødt til at købe små styringsenheder i den udformning, der blev tilbudt af producenten. Hvorfor betale for 16 koblingspunkter, hvis man kun har brug for det halve? Eller hvis styringsopgaverne bliver større? Hvorfor da straks anskaffe sig en komplet udvidelsesenhed?

Slæk ikke på kravene. Hvad enten du "blot" skal udskifte hjælperelær og timere, udføre helt enkle styringsfunktioner eller har regulære styrings- og automatiseringsopgaver.

Ring til os og få tilsendt materiale over 115U.

# 2. Opstilling af udstyret

## Tilslutning

Se afsnit 4 i 100 manual (systemhåndbog)

Adressering

Se afsnit 5 i 100U manual (systemhåndbog)

Opstart

Før de 24 V tilsluttes CPU, stilles netomskifter 0-1 på 0, og RUN-STOP omskifter på STOP. Tilslut spænding og sæt netomskifter på 1. Den gule lysdiode til batteriudfald lyser, hvis der ikke er ilagt batteri. Den røde stoplampe lyser, da CPU står i stop.

## Bemærk:

Kun når CPU er spændingsløs, er det tilladt at isætte/udskifte lagermodul EPROM/EEPROM.

CPU 100U indeholder 1 k RAM brugerprogram.

CPU 102U indeholder 2 k RAM brugerprogram.

CPU 103U indeholder 10 k RAM brugerprogram.

EPROM/EEPROM er program back-up.

Uden batteri isat, vil RAM lageret miste sit program ved strømsvigt. Med batteri isat vil CPU'en bevare sit program ved strømsvigt. Ligeledes bevares status på merkerne indtil merkerbyte 64 (M63.7) og på tællerne Z0 - Z7.

Driftart STOP:

STOP-lampe lyser, program bliver ikke bearbejdet, men status på timere, tællere, merkere og procesafbildning bliver bibeholdt. Alle udgange står på nul. Er der intet batteri, eller er der fejl på dette, lyser gul lampe.

Driftart RUN:

Grøn lampe lyser, program bliver cyklisk bearbejdet, tider, tællere, merkere og indgange bliver bearbejdet, og udgange bliver aktive. Ved overgang fra STOP til RUN bliver de remanente merkere, tællere og procesafbildning resat.

## Bemærk:

Det interne programlager i RAM bliver slettet (kun hvis batteri ikke er isat), hvis der isættes et tomt EPROM/EEPROM lagermodul, og der tilsluttes spænding.

### Driftart COPY

Ved hjælp af COPY funktionen (samme omskifter som RUN/STOP) kan program fra EPROM/EEPROM indlæses i det interne RAM, og fra det interne RAM til EEPROM.

### COPY:

Fra lagermodul til RAM u/batteri - gul lampe lyser. Omskifter 0-1 på 0 (net ud). Modul isættes, omskifter på 1 (net ind) rød lampe lyser, program bliver overført.

Rød lampe blinker, program er for langt eller indeholder ugyldige programblokke. I USTACK bliver bit ASPFA sat, kvitterer med RUN/STOP.

### COPY:

Fra lagermodul til RAM m/batteri - gul lampe slukket. Omskifter 0-1 på 0 (net ud). Modul isættes, COPY taste nedtrykkes og omskifter på 1. (net ind).

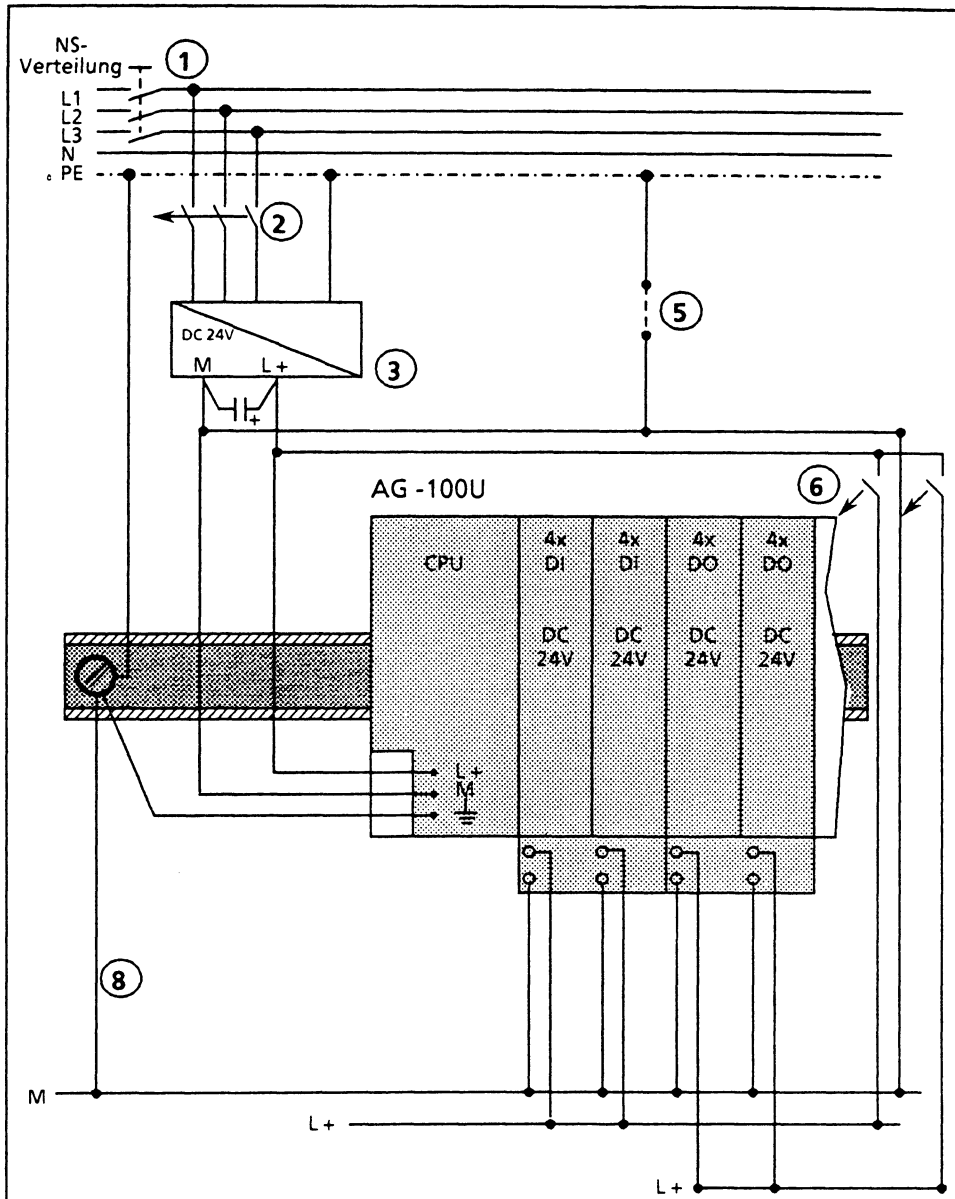
Den røde lampe flimrer, COPY tasten slippes, og program bliver overført. Hvis den røde lampe blinker ved net ind, og COPY tasten slippes, og rød lampe går på konstant lys, så er der intet program (gyldigt) på modulet. Hvis den røde lampe stadig blinker, er programmet for langt eller indeholder ugyldige programblokke.

### COPY:

RAM til lagermodul (kun med batteri).

Omskifter 0-1 på 0 (net ud), EEPROM modul isættes, omskifter på 1 (net ind). COPY tasten nedtrykkes i min. 2 sek., den røde lampe flimrer, COPY tasten slippes, program bliver overført til EEPROM, og den røde lampe lyser konstant. Hvis den røde lampe ikke flimrer, efter at COPY tasten har været nedtrykket i 3 sek., slippes COPY tasten igen.

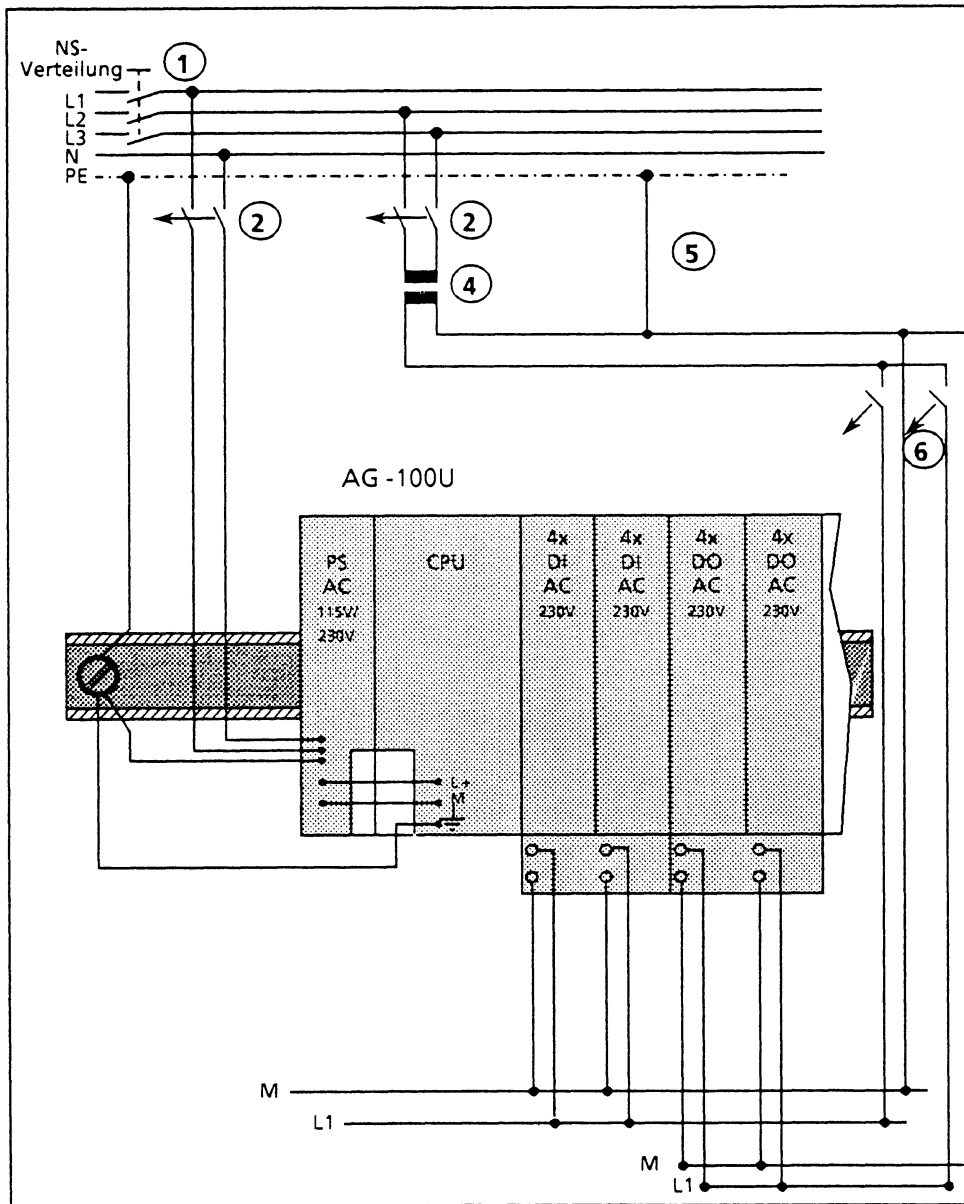
Hvis den røde lampe lyser konstant, er der enten intet modul isat, forkert modul, eller ingen program i RAM lageret. Hvis lampen blinker, er modul defekt, eller der er for lidt lagerplads.



L +: 24 DC  
 M: 0  
 PE: jordskinne  
 N: NUL  
 L1, L2, L3: 3 x 380/220V AC

Eks. på opbygning med 24 V DC styrestrømkreds og laststrøm.





L + 24:DC  
M:0  
PE:jordskinne  
N:0  
L1, L2, L3, : 3 x 380/220V AC

Bemærk: mere detaljeret beskrivelse i afsnit 4 i 100U manual (systemhåndbog).

Eks. på opbygning med 24V DC strømforsyning til CPU.

# Tilslutning af programmeringsenhed

Programmeringsenhed PG 605U tilsluttes på multistikket på forsiden af CPU'en.

## Bestillingsdata

## Bestillingsnr.

Programmeringsenhed PG 605U

i tysk udførelse

6ES5 605-0UA11

i engelsk udførelse

6ES5 605-0UB11\*)

i fransk udførelse

6ES5 605-0UC11\*)

i spansk udførelse

6ES5 605-0UD11\*)

i italiensk udførelse

6ES5 605-0UE11\*)

\*) lagerføres ikke i Danmark

desuden skal der angives bestillingsnr. for brugervejledning.

tysk

6ES5 998-0UP11

engelsk

6ES5 998-0UP21

fransk

6ES5 998-0UP31

dansk

Tastaturskabelon

6ES5 986-0UA11

Transportkuffert

6ES5 986-0LA11

Adapter til printertilslutning

6ES5 984-1UB11

## Anvendelsesområde

Med programmeringsenheden PG605U er der kun mulighed for en driftsform, nemlig: med forbindelse til en PLC. Programmerne kan fremstilles på stedet ved en forbindelse mellem programmeringsenheden og PLC'en (kabellængde 3 m), og de kan overføres til samt testes og korrigeres i RAM-lageret i PLC'en.

# SIMULATORENHED

## Simulatorenhed 788

Simulatorenhed benyttes til simulering af indgangssignaler og til visning af udgangssignaler, hvorfor den er meget velegnet til programtesten. Som alle andre periferienheder monteres den ganske enkelt på busmodulet. Den har dog ingen forbindelse til terminalblokken.

Der kræves ikke 24 V forsyning.

BestillingsdataBestillingsnr.

Simulatorenhed 788

6ES5 788-8MA11

## PROGRAMMERING

Som regel skrives programmet, før det indlæses i PLC. Alt efter brugerens forkundskaber og vaner kan dette ske i form af

- \* en instruktionsliste (AWL),
- \* en funktionsplan (FUP) eller 1)
- \* en nøgleskemaform (KOP). 1)

Programmering med forbindelse til PLC'en (on-line).

Ved denne metode er programmeringsenheden direkte tilsluttet CPU på 100U. Først indlæses programmet trin for trin i det interne programlager i programmeringsenheden ved tryk på tasterne. Dernæst overføres programmet til det interne RAM-lager i PLC.

Hvis programmet ønskes sikret yderligere på et lagermodul, kan dette også ske uden programmeringsenhed. Det er tilstrækkeligt at aktivere tasten "Copy" på centralenheden.

Det er naturligvis også muligt at sikre og indlæse et program på 100U til og fra et isat lagermodul ved hjælp af samtlige programmeringsenheder. Detaljeret beskrivelse heraf - se afsnit 2.

Programmering uden forbindelse til PLC'en (kun med PG 615-635-675-685).

Først indlæses programmet trin for trin via tastaturet i det interne programlager i programmeringsenheden, og dernæst overføres det til et lagermodul EPROM eller EEPROM).

Dernæst indsættes lagermodulet i modulsakten på centralenheden, og programmet indlæses fra lagermodulet i det interne RAM-lager i PLC'en. Det er tilstrækkeligt at holde tasten "Copy" nede, mens der tændes for 100U.

- 1) Kun på programmeringsenheder med skærm - PG 635, PG 675, PG 685, PG 695 eller IBM PC XT/AT.

# 3. Programmering m.v.

## Programmering m.v.

Dette afsnit er ment som en indføring i de grundlæggende funktioner i såvel forståelse af programmeringssproget som i betjening af udstyret.

Hvis du ikke kender STEP 5, som vort programmeringssprog hedder, vil det, inden du går videre, være en god idé at læse lidt om det i bogens afsnit 5.

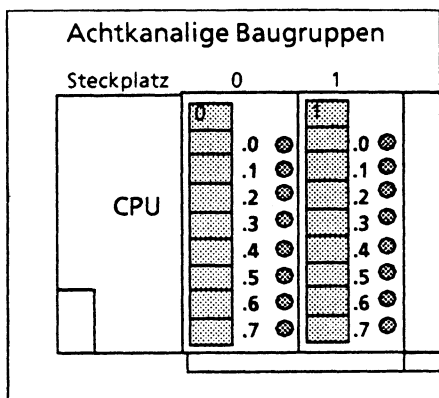
STEP 5 er i sin grundopbygning simpelt og meget let forståeligt. Men det er også fleksibelt og faktisk uden begrænsninger, så man i større PLC-systemer er i stand til at lave komplicerede operationer med en enkelt instruktion (programlinie).

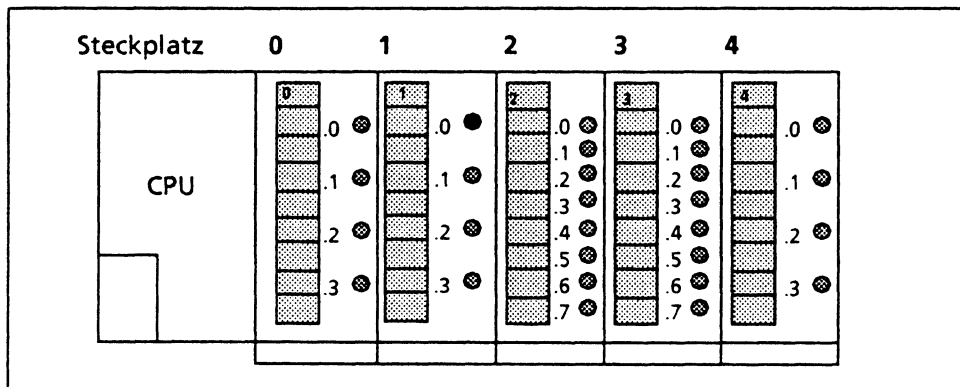
Her i afsnittet behandles de mest grundlæggende funktioner i sammenhænge, der af hensyn til overskueligheden er så simple som muligt. I manualens afsnit om programmering er vist yderligere programmeringseksempler. Du skal naturligvis også forsøge at lave dine egne styringsopgaver eller delopgaver og teste dem på udstyret.

Hvis du på nogen måde under arbejdet med udstyret løber panden mod en mur, skal du blot tage telefonen og ringe til Siemens (tlf. 02 65 65 65). Der vil altid være én til at hjælpe videre.

Tilslut PLC'en og begynd:

Adressering: se adresse beskrivelse i 100U manual (systemhåndbog) afsnit 5.





eks. byte 3 udgang

eks. byte 2 indgang

adresse 3.0 = A 3.0

adresse 2.0 = E 2.0

3.1  
3.2  
3.3  
3.4  
3.5  
3.6  
3.7

2.1  
2.2  
2.3  
2.4  
2.5  
2.6  
2.7

Det er valgfrit, om en plads skal være indgang eller udgang.

E = INDGANG

A = UDGANG

Eksemplerne i dette hæfte er baseret på 8 indgange på plads 0 og 8 udgange på plads 1.

## Opstart af programmering

### Opstart af programmering

Driftsartomskifteren på PLC'en stilles i "stop".

Programmeringsenheden tilsluttes.

Efter ca. 10 sekunder viser display'et:

Hvis det sidste ciffer i øverste linie ikke er 4, fortsættes uden problemer.

Blinkende markør

(cursor)

:	*	P	G	6	0	5	U	V	1	.	4
*	K	O	M	A	N	D	O	*			

Slet øverste linie

Tryk på:



:	⋈										
*	K	O	M	A	N	D	O	*			

# Sletning af lager

## Sletning af lager

Slet eventuelt indhold i programlager.

Slettetastens tekst er hvid. Det betyder, at den hvide skiftetast skal aktiveres først. Når den er aktiveret, er det markeret i display'et med 3 streger ud for teksten "Vortaste".

(Er skiftetasten fejlagtigt aktiveret, trykkes endnu engang på den, og funktionen ophæves).

Tryk først og derefter



hvid skiftetast

:	⌘																		
*	L	O	E	S	C	H	.	A	G	*									

(sletning af PLC)

Kvitter for kommandoen



:	A	G		U	R	L	O	E	S	C	H	E	N		?				
*	L	O	E	S	C	H	.	A	G	*									

Dette er et fortrydelsestidspunkt.  
Herfra er der 2 muligheder.

Fortrydelse:

fører tilbage til udgangspositionen.

:	⌘																		
*	K	O	M	M	A	N	D	O	*										

eller

Udfør sletning  
ved at trykke endnu en gang på kvitteringstasten.



:	*																		
*	K	O	M	M	A	N	D	O	*										

Nu er lageret tomt og klar til at modtage et nyt program.

## Indlæsning

Først forberedes indlæsningen med indlæsetasten. Derefter adresseres programmet til PB 1. 100U har flere programblokke PB, som kan udnyttes, når der er opnået lidt rutine i brugen af sproget.



hvid<sup>↑</sup>skiftetast

:									P	B									1
*	E	I	N	G	A	B	E	*											*

Alle kommandoer og instruktioner afsluttes med kvitteringstasten, som overfører displayvisningen til programmeringsenhedens programlager (RAM-lager).



→	0	0	0	:	*														
P	B																		


000 er første adresse i programlageret, som PG 605U nu er klar til at få indlæst.

# Serieforbindelse

Du starter med at lave en simpel serieforbindelse af 2 sluttekontakter, som er tilsluttet indgangene E 0.0 og E 0.2. Hvis begge kontakter fører "1" signal, bliver udgang A 1.0 aktiveret. Det er "OG"-funktionen (UND), vi benytter.

U E 0 . 0



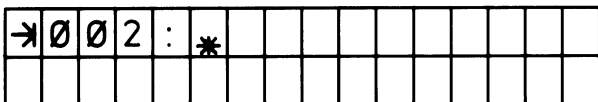
Hvis du taster forkert ind, er dette ikke en katastrofe. Du trykker på tasten til displaysletning , og indlæsefeltet er rent og klar til

fornytt indtastning.

Det korte "OG" eksempel kan diagrammæssigt se ud på flere måder, men programmet er det samme.

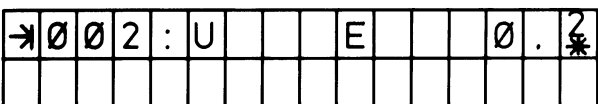
E står for indgang (Eingang)  
A står for udgang (Ausgang)

Kvitter



Næste instruktion:

U E 0 . 2

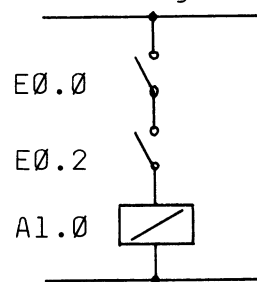


Kvittering

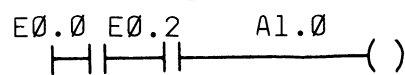


Fremstillingsform for serieforbindelse:

1. Alm. diagram



2. Kontaktplan (KOP)

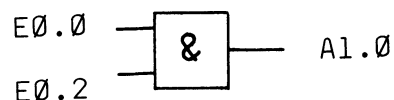


3. Instruktionsliste (AWL)

```

U E 0.0
U E 0.2
= A 1.0
    
```

4. Funktionsplan (FUP) med logiske symboler





Udgang

=	A	1	.	0
---	---	---	---	---

→	0	0	4	:	=			A			1	.	0	*

og kvittering

↕
---

→	0	0	6	:	*									

## Programoverførsel

Det indtastede program befinder sig nu i programmeringsenhedens RAM-lager og skal overføres til PLC'ens RAM-lager: Dette gøres ved at trykke på kvitteringstasten.

↕
---

→	P	G	-	-	>	A	G	?						
P	B		1	*										

Sikkerhedsspørgsmål.

Kvitter igen.

↕
---

:	*													
*	K	O	M	M	A	N	D	O	*					

Før denne displayvisning kommer frem, ses kortvarigt "AKTIV" i display'et.

TILLYKKE!! Nu har DU lavet DIT første program på en Simatic 100U mini-PLC. Held og lykke med de mange følgende programmer!

Men før vi går videre, skal programmet testes.

Driftsartomskifteren på PLC'en sættes på "RUN", og den grønne lampe lyser. Ved at lægge spænding på indgangene E 0.0 og E 0.2 bliver udgang A 1.0 aktiveret. (Signalstatus for ind- og udgange kan ses på de respektive kontrollamper). Har ingen eller kun en af indgangene "spænding", fører udgang A 1.0 intet signal. (0-signal).

I fortsættelse af det indlæste program laves nu en parallelforbindelse af 2 sluttekontakter.

Men forinden skal PLC'en i "STOP", og det indlæste program trækkes over i programmeringsenheden:

## Stop af PLC fra programmeringsenhed

Omskiftning af driftsart kan fjernbetjenes fra programmeringsenheden.

Tast:



:	⌘																		
*	A	U	F	R	U	F	*												



:	⌘																		
*	S	T	O	P		A	G	*											



:	A	G		S	T	O	P		?						
*	S	T	O	P		A	G	*							

Sikkerhedsspørgsmål:

Kvitter igen



:															
*	K	O	M	M	A	N	D	O	*						

Grøn lampe ved driftsartomskifteren slukker, og rød tænder og markerer "Stop".

## Udlæsning af program

Tast følgende ind:



:								P	B						1
*	A	U	S	G	A	B	E	*							

Kvittering



→	0	0	0	:				E			0	.	0		
P	B			1											

Dette er første programlinie.  
Programmet "rulles" frem til sidste instruktion.

Kortvarigt tryk:  
1 instruktion frem.

Konstant tryk:

Programmet "ruller" til sidste adresse.



→	Ø	Ø	6	:	B	E											

Man ser, at "BE" er tilføjet, siden programmet blev overført til PLC'en.  
 "BE" betyder blok-afslutning og føjes automatisk til det indtastede program ved overførsel til PLC'en.

## Parallelforbindelse

Den første instruktion i programtilføjelsen skrives oveni "BE".

Når "BE" vises, kan programmeringen uden videre påbegyndes (uden brug af indlæsetasten).

I en parallelforbindelse anvendes "ELLER" funktionen ("ODER"). På PG 605U er 2 "ELLER" - taster (ODER) "0." og "0". Her skal anvendes "0." - tasten.

Der indtastes følgende

0.	E	0	.	4
----	---	---	---	---

→	Ø	Ø	6	:	0	.		E			Ø	.	4	*



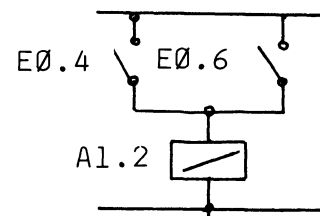
0.	E	0	.	6
----	---	---	---	---

→	Ø	Ø	8	:	0	.		E			Ø	.	6	*

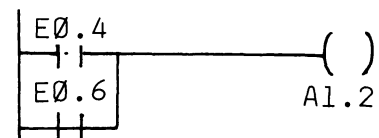


Som for serieforbindelser kan en parallelforbindelse symboliseres på flere måder:

### 1. Almindeligt diagram



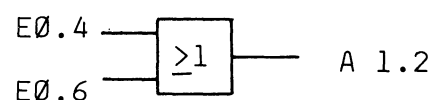
### 2. Kontaktplan (KOP)



### 3. Instruktionsliste (AWL)

O. E Ø.4  
 O. E Ø.6  
 = A 1.2

### 4. Funktionsplan (FUP)



= A 1 . 2

→	Ø	Ø	A	:	=		A				1	.	2	*



→	Ø	Ø	C	:	*									

Så skal programmet overføres til PLC'en.

## Programoverførsel

Tryk på kvitteringstasten:



→	P	G	-	-	>	A	G	?						
P	B		1	*										

Sikkerhedsspørgsmål:

Skal programmet overføres til PLC'en nu?

Ja, kvitter:



→	U	E	B	E	R	S	C	H	R	E	I	B	E	N	?
P	B		1												

Og endnu et sikkerhedsspørgsmål:  
Der står et program i PLC'en. Skal det overskrives  
med et andet?

Ja, kvitter:



:	⌘																		
*	K	O	M	M	A	N	D	O	*										

Programmet er nu overført til PLC'en og skal testes.

## Start af PLC fra programmeringsenhed



:	⌘																		
*	A	U	F	R	U	F	*												



:	⌘																		
*	S	T	A	R	T		A	G	*										



:	A	G		S	T	A	R	T	?										
*	S	T	A	R	T		A	G	*										

Sikkerhedsspørgsmål:



:	⌘																		
*	K	O	M	M	A	N	D	O	*										

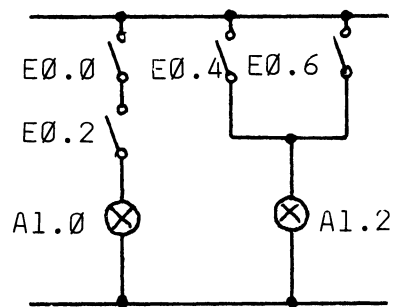
Grøn lampe ved driftartomskifteren lyser og markerer, at PLC'en arbejder.

Programmet testes.

Signal på enten indgang E 0.4 eller E 0.6 eller på begge samtidigt giver signal på A 1.2. Serieforbindelsen af indgangene E 0.0 og E 0.2 med udgangen A 1.0 skal naturligvis stadig fungere.

## Programændringer

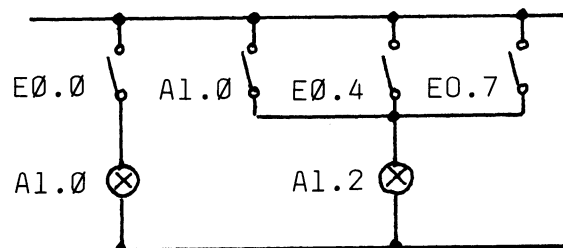
Som nøgleskema ser det indtastede program sådan Ud:



Du prøver nu at udføre 3 forskellige typer af programændringer:

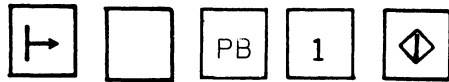
1. Sletning af instruktion.  
E 0.2 skal fjernes
2. Ændring af instruktion.  
E 0.6 skal ændres til E 0.7.
3. Tilføjelse af instruktion.  
Udgang A 1.0 skal være indgang i parallelforbindelsen, så der bliver 3 parallelle sluttekontakter.

Nøgleskemaet ser herefter sådan ud:



Fremgangsmåde:

Programmet trækkes over i programmeringsenheden.



→	0	0	0	:	U			E			0	.	0

1. Programmet køres frem til instruktionen, som fjernes.



→	0	0	2	:	U			E			0	.	2

Der trykkes på skiftetasten og slettetasten



Programlinien slettes, og næste instruktion vises i display'et, men med den sidste adresse.

→	0	0	2	:	U			A			1	.	0

2. Programmet køres frem til instruktionen, som skal ændres.



→	0	0	6	:	U	.		E			0	.	6



Instruktionen overskrives med den nye instruktion (0.E 0.7):

0. E 0 . 7

→	0	0	6	:	0	.	E					0	.	7

Kvittering



Ændring udføres, og display'et viser næste instruktion.

→	0	0	8	:	<del>0</del>	.	A					1	.	2

3. Programmet køres frem til den instruktion foran hvilken, tilføjelsen skal ske.

F.eks.



→	0	0	4	:	<del>0</del>	*	.	E				0	.	4

Instruktionen overskrives med den instruktion, som skal tilføjes (0.A 1.0):

0. A 1 . 0

→	0	0	4	:	0	.	A					1	.	<del>0</del>

Tryk på indføj tasten



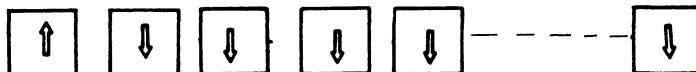
→	0	0	6	:	*	.	E					0	.	4

Instruktionen indføjtes, og display'et viser næste instruktion.

Prøv at øve disse korrektioner nogle gange.

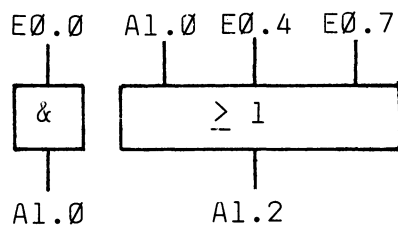
Programændringerne er udført, men inden det ændrede program overføres til PLC'en, kontrolleres programmet i sin helhed.

Programmet "rulles" til 1. instruktion og derefter trinvis frem.



ØØØ:UE      0.0  
 ØØ2:=A     1.0  
 ØØ4:0.A    1.0  
 ØØ6:0.E    0.4  
 ØØ8:0.E    0.7  
 ØØA:=A     1.2  
 ØØC:BE

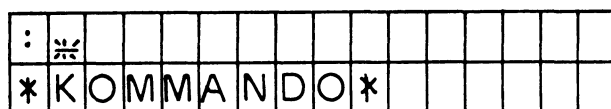
Til orientering kan programmet også tegnes som funktionsplan.



Programændringerne er udført, mens PLC'en har været i drift ("RUN").  
Overførsel af program fra programmeringsenhed til PLC kan ske under drift, men det anbefales at gøre det, når PLC'en er stoppet.

Derfor sættes PLC'en i "STOP" nu, og programmet overføres.

Tryk på kvittering:

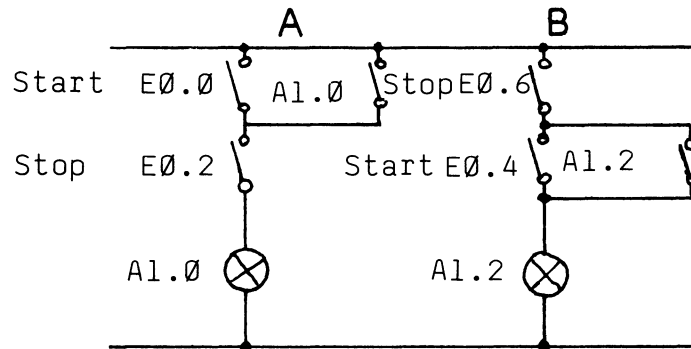
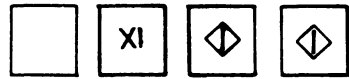


Driftartomskifteren sættes tilbage i "RUN", og programmet testes.

# Start – stop (Hukommelsesfunktion)

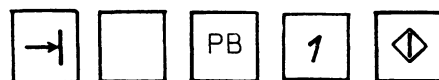
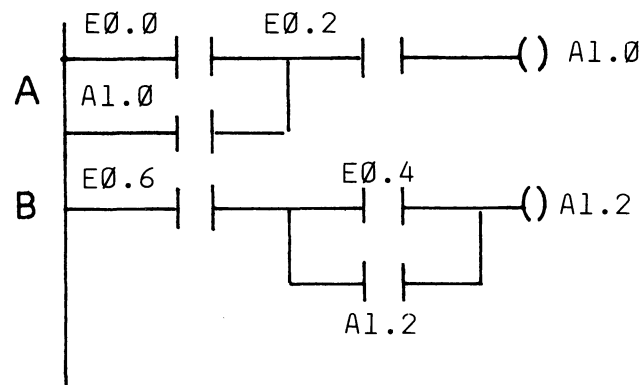
Bring PLC'en i "STOP".

Slet programlageret (se i begyndelsen af afsnittet).



Igen en alternativ tegnemåde, hvor der menes det samme.

Det er lige nemt at programmere ind.








A 1. 0. E 0 . 0 ◊

2. 0. A 1 . 0 ◊

3. U E 0 . 2 ◊

4. = A 1 . 0 ◊

B 5. U E 0 . 6 ◊

6. U( 
7. 0. E 0 . 4 
8. 0. A 1 . 2 
9. ) 
10. = A 1 . 2 

Programmet overføres til PLC'en:



Den bringes i "RUN", og programmet testes.

Ud af ovennævnte program ses, at man kan have en "ELLER" - funktion før en "OG" funktion.

Hvis "OG" funktionen er før "ELLER" kræves, at "ELLER" funktionen sættes i parentes.

Dette eksempel B kan også programmeres ved hjælp af de interne Merkere (hjelperelæfunktioner). Forklaring om merkere er beskrevet lidt længere fremme, men slet igen og prøv.

E E	0.6	Spørgen efter 1 signal (en åben sluttekontakt)
= M	2.0	Mellemresultat 1
0.E	0.4	Spørgen efter "1" signal
0.A	1.2	Holde signal fra udgang 1.2
= M	2.1	Mellemresultat 2
UM	2.0	
UM	2.1	
=A	1.2	

"START-STOP"-funktionen kan også laves med PLC'ens indbyggede hukommelsesfunktioner - "SET" og "RESET" eller "flip-flop'er"


PLC'en bringes i "STOP".

Programmet trækkes ud og "rulles frem til "BE".



Følgende indtastes:

1. U E 0 . 1
2. S A 1 . 1
3. U E 0 . 3
4. R A 1 . 1
5. U E 0 . 5
6. R A 1 . 3
7. U E 0 . 7
8. S A 1 . 3



Her behøves kun et impulssignal for at få selvhold i udgangene A 1.1 eller A 1.3

Overfør program til PLC'en



Sæt PLC'en i "RUN" og test programmet.  
Bemærk ved testen, at den sidst indprogrammerede funktion har dominans: Hvis E 0.1 og E 0.3 er aktiveret samtidig, er der intet signal på udgang A 1.1 (R = RESET har dominans). Hvis E 0.5 og E0.7 aktiveres samtidig, er der signal på udgang A 1.3 (S = SET har dominans).

# Merkere

En merker (internt flag), i PLC-sproget forkortet til M, er en intern funktion, som populært kan sidestilles med et hjælperelæ i en relætavle eller et mellemresultat i programmet. En merker anvendes f.eks. til at fastholde nogle indgangsbetingelser eller funktioner, som skal bruges flere gange i løbet af programmet. Man kan også anvende den til adskillelse af større diagrammer for at lette overskueligheden.

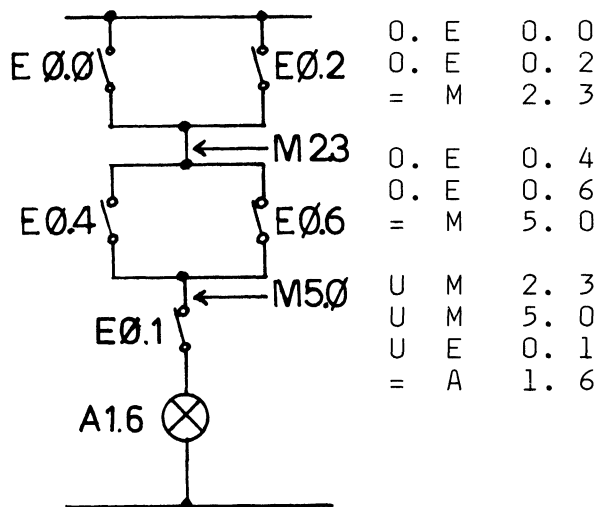
De 1024 merkere i CPU 100 og 102 er grupperet i grupper med 8 i hver. Den første gruppe merkere er 0.0-0.1.-0.2- - 0.7, anden gruppe 1.0 - 1.1 - 1.2- --1.7, og sidste gruppe 127.0 - 127.1 - - - 127.7. Merkerne kan med hensyn til nummerering bruges helt vilkårligt, men man bør starte fra M0.0. I CPU 103 er der 2048 merkere sluttende med M 255.7.

## Specielle merkere

-----

Merkerne M 0.0 - M 63.7 bevarer deres stilling ved spændingsafbrydelse, hvis der er isat pufferbatteri (remanente merkere). Dette kan udnyttes til, at et maskinanlæg fortsætter driften fra den stilling, som det stoppede i, da spændingen forsvandt.

Et eksempel på anvendelse af merkerne kan være:



## Tidsled (T)

16 stk. software timere er indbygget i CPU 100, 32 stk. i CPU 102 og 128 stk. i CPU 103. Til tidsled anvendes programsymbolet T med nummer 0-15. Tidsperioden bestemmes af et tal (højst 999), som angiver antallet af tidsenheder. Dette efterfølges af "." og et tal, 0,1,2 eller 3, som angiver tidsenhedens størrelse:

0	=	10	ms	(0,01 s)
1	=	100	ms	(0,1 s)
2	=	1	S	
3	=	10	S	

Eks: 50.2 = 50 sek.  
40.1 = 4 sek.  
400.0 = 4 sek.









Af hensyn til nøjagtigheden af en tid anbefales at anvende så lille tidsenhed som muligt, da en tids gentagelsesnøjagtighed svarer til tidsenhedens størrelse.

Et program for et tidsled kan se sådan ud, når indgang E 0.0 aktiverer det:

N.B.: Det er tasten med sort "T", der anvendes.

UE 0.0	Start af tidsled
LKT 500.0	Giver et tidsforløb på 5 sek.
SET1	SE=indkoblingsforsinkelse på tidsled nr. 1
UT 1	Der "spørges" på tidsled nr. 1
=A 1.0	Udgangssignal, når tiden er udløbet.
Evt.:	=M X.X, hvis tidsleddet skal anvendes som intern hjælpefunktion.

Og indtastningen, når PLC'en er slettet og klargjort til programmering:

1. U E 0 . 0 
2.  L  K T 5 0 0 . 0 
3.  SE T 1 
4. U T 1 
5. = A 1 . 0 

Når tiden = 5 sek. er udløbet, aktiveres udgangen A 1.0, altså har vi lavet en indkoblingsforsinkelse.  
 Hver gang indgang E0.0 får signal, vil tidsperioden starte forfra, selvom tiden ved afbrydelsen af E 0.0 ikke er udløbet. Hvis du i stedet for tasten "SE" anvender "SI" får du en impuls på udgang A 1.0. Impulsen begynder, når E 0.0 får signal og varer 5 sek.

Altså en signalbegrænsning.

## Taktgenerator

Ved hjælp af 2 tidsled kan der på en meget simpel måde laves en taktgenerator med variabel impuls længde og pause.

```
U E 0.2
UNA 1.4 *)
LKT 500.0
SI T 2
U T 2
= A 1.2 *)
```

```
UNA 1.2 *)
LKT 100.0
SI T 3
U T 3
= A 1.4 *)
```

\*) I stedet for udgange vil man normalt bruge merkere, da en impuls som regel vil være indgang til en tæller eller et blinksignal. Blinksignalet anvendes ofte sammen med et signal, der kan markere en uregelmæssighed, til en alarm enten over blinkende lampe eller horn.

## Udlæsning af status

Før vi går videre med tælleren, vil det være praktisk at se på status-funktionen. Med den kan status på alle programmets elementer udlæses.

Udgangspunkt er displayvisningen KOMMANDO.

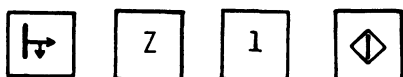
Eks.

↳ E 0 . 6 ↴

* S	T	A	T	U	S	*	E			0	.	6
S	I	G	:	0								

Hvad er signalet på indgang E 0.6?





*S	T	A	T	U	S*	Z							1
KZ	:		1	8									

Hvilken værdi står tæller 1 (Z1) på?

Status kan udlæses i såvel "Stop" som "RUN" for indgange, udgange, merkere, tidsled og tællere.

## Tællere (Z)

En tæller kan tælle både forlæns og baglæns. Tælleimpulsen kan være en indgang, en merker eller en udgang.

Den simpleste form for en tæller ser sådan ud:

```

UE 0.0   tælle impuls
ZVZ1    tælle fremad på tæller nr. 1
UZ1     spørge tæller 1
=A 1.0   tællerudgang ( evt. som merker)

```

Prøv at lave denne tæller og udlæs status på den! (Se forrige side).

Når tælleren har en værdi forskellig fra 0 vil A 1.0 give signal.

Det er ikke til stor nytte, men der kan hægtes en hel del "ekstraudstyr" på.

Dette komplicerer naturligvis tælleren, men så kan den til gengæld alt, hvad man kan ønske sig af en tæller: Tælle forlæns, tælle baglæns, nulstilling, reset til en eller flere indprogrammerende startværdier, give signal, når tælleren er under eller over en eller flere givne værdier og give signal over flere forskellige intervaller.

PLC'en har 2 lagerceller (akkumulatorer), hvor der kan være en værdi (tal) i hver. Når der indtastes "L" + en operand f.eks. KF 10 (heltal 10), får akkumulator 1 en værdi på 10. Hvis der derefter indtastes en ny lade-instruktion, flyttes 10-tallet over i akkumulator 2, og den nye værdi lægges i akkumulator 1.

Slet PLC'en og indprogrammer:

```

U E 0.0 } tælleren starter på værdien 10
LKZ 10  }
S Z 1   }
U E 0.2 } tæller fremad
ZVZ 1   }

U E 0.4 } tæller baglæns
ZR  Z1  }
U E 0.6 } stiller tælleren på 0
R Z 1   }
L Z 1   }
L KF 20 } tællerens værdi og tallet 20 lades
= F     } over i akkumulatoren og sammen-
S A 1.6 } lignes. Hvis tællerens værdi er > 20,
L Z 1   } sættes A 1.6
LKF 19  } tællerens værdi og tallet 19 lades
= F     } over i akkumulatoren og sammen-
R A 1.6 } lignes. Hvis tællerens værdi < 19,
        } resettes A 1.6

```

(Signal på A 1.6, når  $Z1 \geq 20$ )

Programmet kan fortsættes med at lade andre tælle-  
værdier + tal, sammenligne dem, sette og resette  
den samme eller andre udgange.

F.eks.

```

L Z 1 }
LK F 24 }
= F     }
R A 1.7 } Signal på A 1.7, når Z1 > = 25
L Z 1   }
LK F 25 }
= F     }
S A 1.7 }
L Z 1   }
LK F 30 }
= F     }
S A 1.7 } Signal på A 1.7, når Z1 < = 30
L Z 1   }
LK F 31 }
= F     }
R A 1.7 }

```

(=Signal på A 1.7, når  $25 \leq Z1 \leq 30$ )

Overfør programmet til PLC'en og foretag test.

Når du udlæser status på Z1, kan du hele tiden  
følge med i, hvad der sker med tælleren.

Hvis du kun har brug for at "sette" en udgang  
eller merker, når tælleren er over en vis værdi,  
er det mere praktisk at skrive:  $>F$   
= AX.X

i stedet for at sammenligne på lighed.

## Ændring af timere og tælleres værdier under drift

PLC 100U giver sammen med PG 605U mulighed for at ændre på timeres og tælleres indstillingsværdier under drift og uden at bryde ind i programmet.

Værdierne for timerne lagres i dataordene 0-15 (31 og 127 for 102 h.h.v. 103) i datablok nr. 11 (DW 0-15 i DB 11).

Værdierne for tællere lagres i dataordene 0-15 (31 og 127 for 102 h.h.v. 103) i datablok 12.

Dataordets nummer refererer direkte til en tæller (DW 0-15 i DB 12) eller timer med samme nummer. Det vil sige, at man kan have variabel værdi på både timer T5 og tæller Z5, men værdierne ligger i hver sin datablok.

Programmet for en timer vil med variabel tidsværdi se sådan ud:

```
A DB 11 åbning af datablok 11
UE 0.0
L DW 1 dataord nr. 1 lades ind i timer
SET 1
UT 1
= A 1.0
```

Linien A DB 11 skal indtastes på følgende måde:

\*  1  3 +  1  1  ↕

Linien L DW 1 skal indtastes på følgende måde:

\*  0  0 +  1  ↕

hvilket giver teksten på displayet.

"1" - tallet til sidst giver dataordets nummer og kan være 0-15 afhængigt af timer/tællernummeret.

ADVARSEL: Programmet må ikke sættes i drift, før der er oprettet en DB 11 med de anvendte dataord. Ellers vil PLC ikke gå i RUN. De aktuelle timer/tællerværdier kan indlæses direkte i DB 11 under drift med PG 605U eller OP 393 operatørenhed.

## PROGRAMMERING AF DATABLOK

En datablok er et lagerareal, hvor der ikke sker nogen programbearbejdning. Det er kun et læse- og skriveareal. Det vil sige, at man kan overføre (Transferere = T) en værdi (tid, tal o.lign.) til et dataord (DW) i en datablok for senere og måske i anden sammenhæng at hente (Læse=L) den igen.

Forberedelse af ind-udlæsning

Forberedelse af ind- og udlæsning finder sted som ved de andre datablokarter. I stedet for FB, OB eller PB indlæses

Indlæsning  → \*  1  1  ↕

Udlæsning  ⇨ \*  1  1  ↕

Der kan indlæses i følgende formater:

KH (konstant hexadecimal)

KF (konstant heltal)

KT (konstant tidsværdi)

KZ (konstant tællerværdi)



Skift af format

## Indlæsning af data



Ønsket format indstilles

Eksempel: Format KT er indstillet.

Indtast nu blot tidsværdien f.eks. 100.1 ved de dataord, som kaldes fra timerne. Øvrige dataord indtastes med værdien KT 0.0. Efter det sidst indtastede dataord overføres datablok 11 til PLC'en med kvitteringstasten. Datablokken skal have en længde, som er lig med eller større end det timer-nr., som er højest.



Overførsel henholdsvis overskrivning

Efter DB 11 er overført, sættes CPU RUN. Overraskelsen her er, at der intet sker. Det skal der heller ikke.

Nu er vores program bygget op af flere blokke, nemlig PB1 og DB 11.

Det opfatter CPU'en som en strukturering af programmet, hvilket vil sige, at den kræver Organisationsblok 1 - OBI-indlæst.

I OBI skal den overordnede strukturering af programmet ligge. Eller sagt på en anden måde: Fra OBI opkaldes de enkelte programafsnit, PB'er, FB'er og DB'er. Der er 64 stk. til rådighed af hver (med numrene PB0-63 og DB2-63) i CPU 100 og 102. I CPU 103 kan blokke nummereres med op til nr. 255.

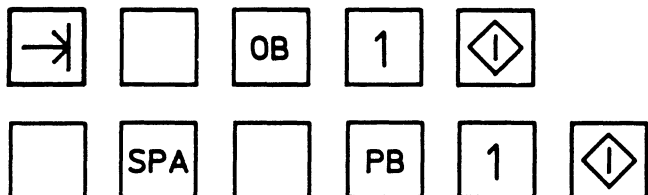
Et typisk program i OBI ser således ud:

```
SPA PB 1
SPA PB 3
SPA PB 40
SPA FB 17
U E 0.0
SPB PB 7
SPA FB 4
U E 0.1
SPB FB 10
o.s.v.
```

Blokkene, som står efter en "SPB" bearbejdes kun, hvis den foranstående betingelse er "1". Når betingelsen bliver "0" stopper bearbejdningen af blokken men, Vigtigt: Status bibeholdes, d.v.s. hvis en udgang er "1", forbliver den "1".

Behandling af blokkene, som står efter "SPA" er mere enkel. De bearbejdes hele tiden.

For at få vort program til at køre indtastes i 0B1:  
"SPA PBl"



og overføres til PLC'en.

Herefter virker timeren.

Når tidsværdien skal ændres under drift, trykker du på status-tasten, og "T1" og kvitterer.

I displayet vises

*	S	T	A	T	U	S	*	T		1		
K	T	:			0	.	1					

("1" er tidsbasen, og "0" er timerens aktuelle værdi - den er altså udløbet)

Skal værdien ændres, trykkes en gang mere på status-tasten, og displayet viser:

*	S	T	A	T	U	S	*	T		1		
D	B		1	*								

DB-nummeret kan ændres her, hvis du har valgt en anden DB til tidsværdierne. Dette gøres ved indtastning af nyt nr.

Kvittering



*	S	T	A	T	U	S	*	T		1			
K	T	:	1	0	0	.	1						

KT 100.1 er den tidsværdi, som du programmerede ind i dataord 1 datablok 11. Cursoren står under tallet, og en ny værdi kan indtastes, f.eks. 200.1.

2	0	0	.	1
---	---	---	---	---



Når du kvitterer, overføres den nye værdi til dataord 1. Den bruges næste gang, timeren startes.

Såvel PBl som OBl og DB11 kan kopieres til EEPROM-lagermodul - se side 2.2. Hvis der ikke er batteri i PLC'en, vil tidsværdien efter spændingsindkobling altid være den oprindelige værdi, som ligger på lagermodulet. Hvis der er batteri i PLC'en, vil tidsværdien efter spændingsindkobling altid være den sidst indtastede værdi.

## Programkopiering:

Foreløbig har vi arbejdet med program i CPU'ens RAM-lager. Et RAM-lager er et flygtigt lagermedium, d.v.s hvis der ikke er batteri-back-up, og netspændingen forsvinder, går programmet tabt.

For at hindre den situation kan programmet overføres til et lagermodul med EEPROM:

### Bemærk:

Kun når CPU er spændingsløs, er det tilladt at isætte/udskifte lagermodul EPROM/EEPROM.

Fremgangsmåden er følgende (batteri skal være isat):

1. Sæt lagermodulet i CPU'ens skakt og tænd for spændingen.
2. Driftsartomskifteren trykkes på "COPY" (fjederpåvirket) og holdes, indtil rød lampe begynder at blinke.
3. Omskifteren slippes.
4. Når den røde lampe holder op med at flimre og går på fast rødt lys, er kopieringen sket.
5. Omskifteren kan sættes på RUN igen.

Kopiering fra lagermodul til CPU's RAM-lager:

- A. Uden batteri-back-up.

Med isat lagermodul kopieres programmet automatisk over, når forsyningsspændingen tilsluttes.

- B. Med isat batteri

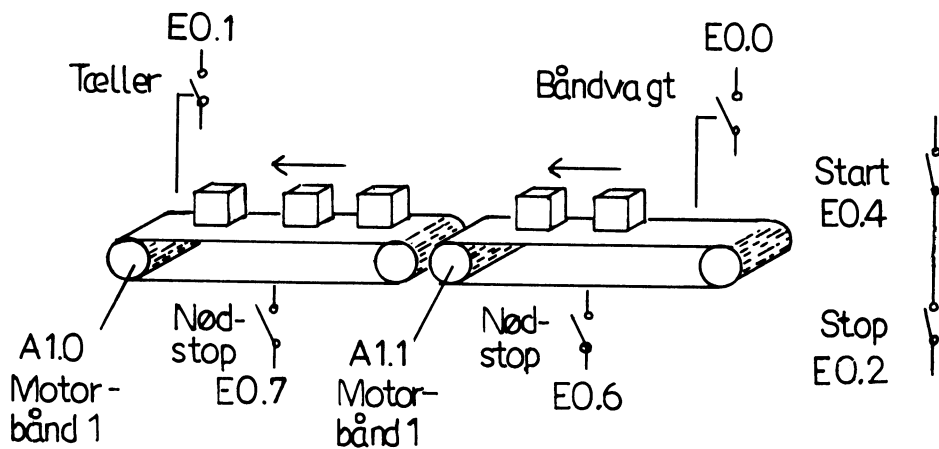
1. Lagermodul isættes.
2. Driftsartomskifteren holdes på "COPY" samtidig med, at forsyningsspændingen tilsluttes.
3. Når den røde lampe flimrer, slippes omskifteren.
4. Når den røde lampe går på fast lys, er kopieringen sket.
5. Driftsartomskifteren kan sættes på RUN, og CPU'en behandler det nye program.

NB: Specielt for CPU 102

Programmet afvikles først med hurtig bearbejdningstid, når en programkopiering til lagermodul har fundet sted.



## Opgave:



Her vises et eksempel på en mindre båndstyring.

Start af bånd:	Trykkontakt	E 0.4
Stop af bånd:	Trykkontakt	E 0.2
Stop af bånd:	Nødstop	E 0.6
Stop af bånd:	Nødstop	E 0.7

Bånd 1 kan startes med trykkontakten E 0.4, hvis bånd 2 står stille (E 0.0 sluttet). 5 sek. efter bånd 1 er startet, starter bånd 2. Når 20 kasser har passeret tælleren, skal begge bånd stoppe, og tælleren nulstilles. Båndene kan genstartes af E 0.4.

Forsøg at udarbejde styringsprogram for båndene. Programmet indeholder almindelige "=" funktioner, "set" og "reset", tidsled, tæller og sammenligningsfunktion. På de næste to sider er fortrykt instruktionslister, som kan anvendes til programudarbejdning og til programdokumentation. Bagefter er vist en mulig løsning af opgaven.

Linier	STEP-5-instruktion			Bemærkninger
	Operation	Symbol	Operand Parameter	
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
A				
B				
C				
D				
E				
F				
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
A				
B				
C				
D				
E				
F				
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
A				
B				
C				
D				
E				
F				
<b>SIEMENS</b>	Automationssystem Simatic			Initialer
				af

Linier	STEP-5-instruktion			Bemærkninger
	Operation	Operand		
		Symbol	Parameter	
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
A				
B				
C				
D				
E				
F				
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
A				
B				
C				
D				
E				
F				
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
A				
B				
C				
D				
E				
F				
<b>SIEMENS</b>	Automationssystem Simatic			Initialer
				af

LØSNINGSFORSLAG

## STEP-5-instruktion

<u>Opera- tion</u>	<u>Sym- bol-</u>	<u>Para- meter</u>	<u>Bemærkninger</u>
U	E	0.0	Klart signal fra bånd 2
U	E	0.4	Start tryk impuls
S	M	3.0	Merker sat (selvhold)
O	E	0.2	Stop
O	E	0.7	Nødstop
O	E	0.6	Nødstop
O	M	1.0	Udløb tæller
R	M	3.0	Reset merker
U	M	3.0	Spørgen på merkersignal
=	A	1.0	Start bånd 1
U	A	1.0	Spørgen på bånd 1
L K	T	500.0	Lader konstant tid 5 sek.
S E	T	1	Set på timer 1
U	T	1	Spørgen på timer
=	A	1.1	Udgang bånd 2
U	A	1.1	Spørgen om bånd 2 kører
U	E	0.1	Tælle puls
Z V	Z	1	Tælle forlæns på tæller 1
L	Z	1	Lade tæller 1
L K	F	20	Konstant faktor 20
= F			Sammenligning
=	M	1.0	Når tællerværdi er 20
R	Z	1	Reset tæller

## Øvrige danske håndbøger

Denne programmerings-ABC afsluttes her. Vi håber, at den har gjort dig dus med SIMATIC 100U mini-PLC og programmeringsenhed PG 605, så du har fået mod på at gå i gang med et større samlet program.

Af yderligere dokumentation foreligger følgende:

SIMATIC 100U Brugsanvisning/programmeringsvejledning (tysk eller engelsk). 1 stk. følger med CPU ved køb af SIMATIC 100U CPU. Katalog ST 52.1 og SIMATIC S5 605U brugsanvisning på dansk kan rekvireres.

Ovennævnte anbefales som håndbøger i forbindelse med almindelig brug af udstyret, og disse beskriver i fuldt omfang mulighederne, som tilbydes af 100U henholdsvis 605U.

SIMATIC 100U er en af en større serie af PLC'er, som kan programmeres af forskelligt programmeringsudstyr, og derfor er programmeringsvejledningen naturligvis en universal-vejledning. Det samme gør sig gældende med hensyn til programmeringsenheden PG 605U, som kan anvendes i forbindelse med programmering, idriftsætning og fejlfinding på flere U-række PLC'er. Derfor er der på PG 605U nogle funktioner, som kun er anvendelige i forbindelse med de større PLC'er.

### Programdokumentation

Det færdige program kan skrives ud på printer ved at indskyde en printeradapter mellem CPU og stik på programmeringsenheden.

Bestillingsdata:

Printeradapter                      Best.nr. 6ES5 984-1UB11

Der skal tilsluttes printer med TTY 20 mA interface med indstilling på 300 eller 600 baud. Desuden skal printeren have strømforsyning til datatransmission, f.eks. Siemens type PT88N/TTY nåleprinter eller type PT 88I/TTY jet ink-printer.

Hvis programmerne ønskes dokumenteret i nøgleskema- eller funktionssymbolform, skal dette ske på en af de større programmeringsenheder, f.eks. PG 675. Men det kræver desuden, at programmet "skilles ad" i strømveje af kommandolinien \*BLD255, som indtastes: skift-\*-08(\*BLD) + 255. Denne kommando indføjes efter hver strømvej eller afsluttet logikblok. Desuden skal timere og tællere forøges med 2,3 eller 4 instruktioner NOPO, som indtastes: skift-\*-09. Yderligere materiale herom kan rekvireres.

# 5. Programmeringsproget STEP-5

Programmeringen af SIMATIC 100U sker i det simple PLC-sprog:

STEP 5.

STEP 5 kan skrives (fremstilles) på 3 måder:

1. Instruktionsliste, som består af en række memotekniske forkortelser
2. Kontaktplan, som består af kontaktsymboler for slutte- og brydekontakter, som det kendes fra nøglediagrammer.
3. Funktionsplan, (logikdiagram) som består af blokke til de logiske funktioner: OG, ELLER, o.s.v.

Eksempler:

UE Ø.2  
=A 1.Ø

E Ø.2  
—| |— ( )A 1.Ø

E Ø.2  
— [ & ] — A 1.Ø

Til programmering med PG 605U er kun instruktionsliste anvendelig, og vi ser bort fra kontaktplans- og funktionsplansprogrammering.

# Opbygning af en instruktion

En instruktion er den mindste del af STEP 5 programmet.

Den består af:

Operation - "hvad skal gøres?" og

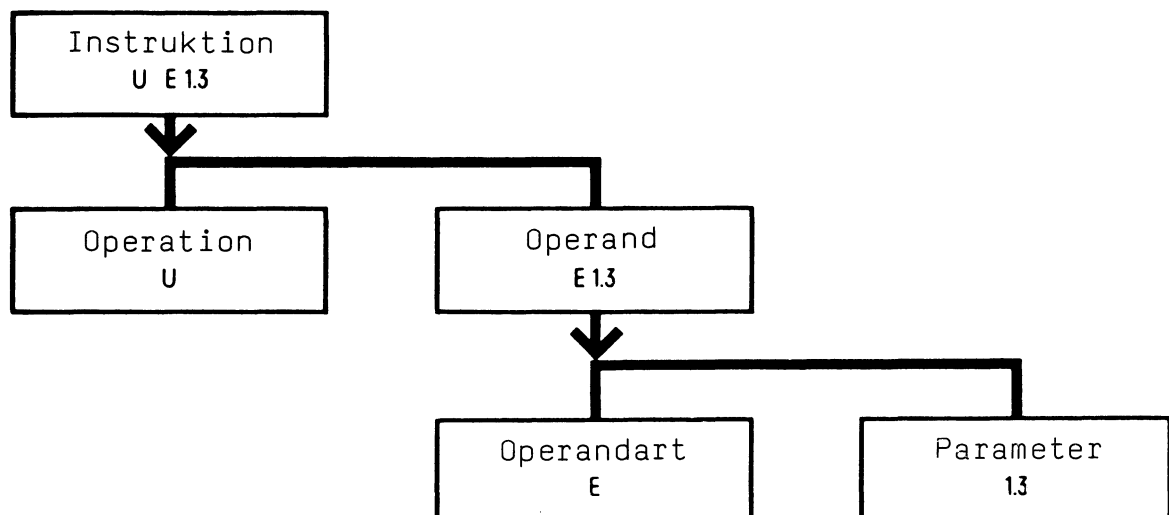
Operand - "hvad skal det gøres med?"

Operanden kan deles i to:

Operandarten (indgang, udgang, merker o.s.v.)

Parameter.

Parameteren angiver nummeret på indgangen, udgangen o.s.v., som forespørges af instruktionen.



PLC'en "spørger" om signalstatus på sine indgange og behandler dem svarende til instruktionslisten. Denne behandling giver et facit - forbindelsesresultatet - fremover forkortet til VKE.

## OG-forbindelse

OG-forbindelsen svarer til en serieforbindelse af kontakter. Forbindelsesresultatet er "1", når alle kontakter har signalstatus "1".

1. Signal	2. Signal	Resultat (VKE)
0	0	0
0	1	0
1	0	0
1	1	1

## ELLER-forbindelse

ELLER-forbindelsen svarer til en parallelforbindelse af kontakter. Forbindelsesresultatet er "1", når en af kontakterne har signalstatus "1".

1. Signal	2. Signal	Resultat (VKE)
0	0	0
0	1	1
1	0	1
1	1	1



# Betydning af operarandart

E: Indgang

A: Udgang

Som operander tillades:

E 0.0 - E 0.7            A 0.0 - A 0.7

o.s.v. se mærkning på de aktuelle komponenter

M: Merker

Som operander tillades:

M 0.0.....M 0.7

M 1.0.....M 1.7

.

.

.

M 127.0.....M 127.7 for CPU 100 og 102

M 255.0.....M 255.7 for CPU 103

T: Tidsled (timer)

Z: Tæller

som operander tillades:

TO til T15/T31/T127    Z0 til Z15/Z31/Z127

MB: Merkerbyte

EKS.: MB 4 = M 4.0 til M 4.7

KT: konstant tidsværdi

som operander tillades:

KT 0.0 til KT 999.0            0 til 999 x 0,01 s

KT 0.1 til KT 999.1            0 til 999 x 0,0 s

KT 0.2 til KT 999.2            0 til 999 x 1 s

KT 0.3 til KT 999.3            0 til 999 x 10 s

KZ: Konstant tællerværdi

som operander tillades:

KZ 0 til KZ 999

KF: Konstante fastpunkttal

Som operander tillades:

KF 0 til KF 32 767

# Oversigt over operander

Operandmærke	Betydning	Parameterområde		
		CPU 100	CPU 102	CPU 103
A	Udgang i (I/O-kopiering)	0,0 til 31,7		0,0 til 127,7
AB	Udgangsbyte (i I/O-kopiering)	0 til 31		0 til 127
AW	Udgangsort (i I/O-kopiering)	0 til 30 (64 til 127 for analog udlæsning)		
BF	Bytekonstant (fast punkt)	-127 til +127		
D	Bit i dataord	0,0 til 255,15		
DB	Datablok	2 til 63		2 til 255
DL	Venstre byte i dataord	0 til 255		
DR	Højre byte i dataord	0 til 255		
DW	Dataord	0 til 255		
E	Indgang (i I/O-kopiering)	0,0 til 31,7		0,0 til 127,7
EB	Indgangsbyte (i I/O-kopiering)	0 til 31		0 til 127
EW	Indgangsort (i I/O-kopiering)	0 til 30 (64 til 127 for analog indlæsning)		
FB	Funktionsblok	0 til 63		0 til 255
KB	Konstant "1 byte tal"	0 til 255		
KC	Konstant "karakter" (2 tegn)	ASCII-tegn		
KF	Konstant "heltal"	-32768 til +32768		
KH	Konstant "hexadecimaltal"	0 til FFFF		
KM	Konstant "bitmønster"	16 bit		
KT	Konstant "tidsværdi" (BCD-koderet)	0,0 til 999,3		
KY	Konstant "2 byte tal"	0 til 255 (pr.byte)		
KZ	Konstant "tællerværdi" (BCD-koderet)	0 til 999		
M	Merker			
	- remanente	0.0 til 63.7		
	- ikke remanente	64.0 til 127.7		64.0 til 255.7
MB	Merkerbyte	0 til 127		0 til 255
MW	Merkerord	0 til 126		0 til 254
OB	Organisationsblok	1,21,22,34	1,21,22,34	1,2,13,21
	Programblok (ved SPA,SPB)	0 til 63		0 til 255
PB	Periferibyte			0 til 127
PW	Periferiord			0 til 126
SB	Skridtblokke			0 til 255
T	Timere	0 til 15	0 til 31	0 til 127
Z	Tællere	0 til 15	0 til 31	0 til 127
	- heraf remanente	0 til 7	0 til 7	0 til 7

# Operationsoversigt STEP 5 - Grundoperationer

Operation			Operandmærke (tilladt)			Bearbejdnings-tid i $\mu$ s		Beskrivelse		
AWL	FUP	KOP	Fremstilling som instruktions-liste	funktionsplan (ikke muligt med PG 605U og PG 615)	nøgleskemaf.	CPU 100 CPU 102		AWL	FUP	KOP
<b>Forbindelsesoperationer</b>										
U			E	A	M	T	Z	70	Spørgen efter signaltilstand „1“ og forbindelse efter OG	Serieforbindelse af sluttekontakter
UN			E	A	M	T	Z	75	Spørgen efter signaltilstand „0“ og forbindelse efter OG	Serieforbindelse af brydekontakter
O			E	A	M	T	Z	75	Spørgen efter signaltilstand „1“ og forbindelse efter ELLER	Parallel forbindelse af sluttekontakter
ON			E	A	M	T	Z	80	Spørgen efter signaltilstand „0“ og forbindelse efter ELLER	Parallel forbindelse af brydekontakter
O								41	ELLER-forbindelse af OG-funktioner	Tilbageførsel af strømbane som parallel tilslutning
U(								61	OG-forbindelse af parentesudtryk (6 parentesniveauer)	Åbning af en forgrening
O(								64	ELLER-forbindelse af parentesudtryk (6 parentesniveauer)	Tilbageførsel og åbning af en forgrening
)								51	Afsluttende parentes (afslutning af et parentes-udtryk)	Lukning af en forgrening
<b>Hukommelsesoperationer</b>										
S			E	A	M			70	Settes ved resultat „1“; ingen virkning ved „0“	
R			E	A	M			70	Resettes ved resultat „1“; ingen virkning ved „0“	
=			E	A	M			70	Tildelt signal „1“ ved resultat „1“; Tildelt signal „0“ ved resultat „0“	
<b>Indlæsningsoperationer</b> (indholdet af operanden eller værdien af konstanten indlæses i registret – et mellem-lager)										
L			EB	AB				61		Indlæsning af en byte fra I/O-kopieringen
L			EW	AW				61		Indlæsning af et ord fra I/O-kopieringen
L	Disse instruktioner skal indlæses som AWL.	Disse instruktioner skal indlæses som AWL.	MB					64		Indlæsning af en merkebyte
L			MW					71		Indlæsning af et merkeord
L			DL					65		Indlæsning af den venstre byte af et dataord fra den aktuelle datablok
L			DR					65		Indlæsning af den højre byte af et dataord fra den aktuelle datablok

Mere detaljeret oversigt findes i manual, afsnit 13.

# Operationsoversigt STEP 5 - Grundoperationer

Operation			Operandmærke (tilladt)	Bearbejdnings- tid i $\mu$ s	Beskrivelse	
Fremstilling som instruk- tions- liste	funktions- plan	nøgleskema- form				
AWL	FUP	KOP		CPU 100 CPU 102		
<b>Indlæsningsoperationer (fortsat)</b>						
L			DW	66	Indlæsning af et dataord fra den aktuelle datablok	
L	Disse instruktioner skal indlæses som AWL	Disse instruktioner skal indlæses som AWL	KB <sup>1)</sup>	54	Indlæsning af en konstant (1-byte-tal)	
L			KC <sup>1)</sup>	KF	57	Indlæsning af en konstant (karakter, heltal)
L			KH <sup>1)</sup>	KM <sup>1) 2)</sup>	57	Indlæsning af en konstant (hexadecimaltal, bitmønster)
L			KY <sup>1)</sup>	KT	57	Indlæsning af en konstant (2-byte-tal, tidsværdi)
L			KZ		57	Indlæsning af en konstant (tællerværdi)
L			T	Z	70	Indlæsning af indholdet af en timer eller tæller (dual-kodet)
LC			T	Z	125	Indlæsning – BCD-koderet – indholdet af en timer eller tæller
<b>Overførselsoperationer (indholdet af registret overføres til operanden)</b>						
T	Disse instruktioner skal indlæses som AWL	Disse instruktioner skal indlæses som AWL	EB	AB	52	Overførsel af en byte til I/O-kopiering
T			EW	AW	54	Overførsel af et ord til I/O-kopiering
T			MB		55	Overførsel af en byte til merkerne
T			MW		64	Overførsel af et ord til merkerne
T			DL		53	Overførsel af en byte til venstre i et dataord i den aktuelle datablok
T			DR		57	Overførsel af en byte til højre i et dataord i den aktuelle datablok
T			DW		59	Overførsel af et ord til et dataord i den aktuelle datablok

1) Ikke muligt med PG 605U.

2) Ikke muligt med PG 615.

Mere detaljeret oversigt findes i manual, afsnit 13.

# Operationsoversigt STEP 5 - Grundoperationer

Operation			Operandmærke (tilladt)	Bearbejdnings- tid i $\mu$ s	Beskrivelse
Fremstilling som instruk- tions- liste	funktionsplan (ikke muligt med PG 605U og PG 615)	nøgleskemaf.			
AWL	FUP	KOP			
<b>Timeroperationer</b> (værdien for en tid, der skal begyndes, skal på forhånd indlæses i registret; VKE = forbindelsesresultat)					
SI			T	125	Start af en impulstimer ved positivt signalkift for VKE (signalbegrænsning); ved VKE „0“ går timerudgangen på „0“.
SV			T	125	Start af en timer som forlænget impuls ved positivt signalkift for VKE (signalbegrænsning og -forlængelse); ved VKE „0“ forbliver timerudgangen uændret.
SE			T	127	Start af en indkoblingsforsinket timer ved positivt signalkift for VKE; ved VKE „0“ går timerudgangen på „0“.
SS			T	127	Start af en indkoblingsforsinket timer med hukommelse ved positivt signalkift for VKE; ved VKE „0“ forbliver timerudgangen uændret.
SA			T	66	Start af en udkoblingsforsinket timer ved negativt signalkift for VKE; ved VKE „1“ går timerudgangen på „0“.
R			T	67	Reset (nulstilling) af en timer ved forbindelsesresultat „1“, ingen virkning ved „0“.
<b>Tælleroperationer</b>					
ZV			Z	79	Tælling fremad af en tæller med 1 ved positivt signalkift for forbindelsesresultatet.
ZR			Z	92	Tælling tilbage af en tæller med 1 ved positivt signalkift for forbindelsesresultatet.
S			Z	118	Preset af en tæller ved positivt signalkift for forbindelsesresultat; værdien skal først indlæses i akkumulatoren; ingen virkning ved „0“.
R			Z	69	Reset (nulstilling) af en tæller ved forbindelsesresultat „1“, ingen virkning ved „0“.
<b>Sammenligningsoperationer</b> (for heftal; resultatet af den seneste operation står altid i akkumulatoren). Desuden bliver VKE = „1“ ved sandt udsagn og „0“ ved falsk udsagn.					
! = F				79	lig med; akkumulator 2 lig med akkumulator 1
> < F				82	forskellig fra; akku 2 er forskellig fra akku 1
> F				79	større end; akku 2 er større end akku 1
> = F				79	større end eller lig med; akku 2 er større end eller lig med akku 1
< F				82	mindre end; akku 2 er mindre end akku 1
< = F				82	mindre end eller lig med; akku 2 er mindre end eller lig med akku 1

Mere detaljeret oversigt findes i manual, afsnit 13.

# Operationsoversigt STEP 5 · Grundoperationer

Operation			Operandmærke (tilladt)	Bearbejdnings- tid i $\mu$ s	Beskrivelse
Fremstilling som instruk- tions- liste			funktionsplan (ikke muligt med PG 605U og PG 615)	nøgleskemaf.	
AWL	FUP	KOP			
<b>Regneoperationer</b> (resultatet af den seneste operation står altid i akkumulatoren)					
+	Disse instruktioner skal indlæses som AWL	Disse instruktioner skal indlæses som AWL		55	Sammenlægning af heltal; akku 2 + akku 1, resultat står i akku 1
-	Disse instruktioner skal indlæses som AWL	Disse instruktioner skal indlæses som AWL		58	Subtraktion af heltal; akku 2 - akku 1, resultat står i akku 1
<b>Blokkaldoperationer</b>					
SPA			FB	147	Absolut kald af en funktionsblok
SPB			FB	152	Betinget kald af en funktionsblok
SPA	Disse instruktioner skal indlæses som AWL	Disse instruktioner skal indlæses som AWL	PB	125	Absolut kald af en programblok
SPB	Disse instruktioner skal indlæses som AWL	Disse instruktioner skal indlæses som AWL	PB	130	Betinget kald af en programblok
A	Disse instruktioner skal indlæses som AWL	Disse instruktioner skal indlæses som AWL	DB	70	Åbning af en datablok
<b>Blokafslutningsoperationer</b>					
BE	BE	BE		88	Blokafslutning (sidste operation i en blok)
BEB	BEB	BEB		90	Betinget blokafslutning
BEA	BEA	BEA		88	Absolut blokafslutning (inden for en blok, dvs. kun i funktionsblokke)
<b>Nul-, stop-, billedopbygningsoperationer</b>					
NOP 0	Disse instruktioner skal indlæses som AWL	Disse instruktioner skal indlæses som AWL		35	Nuloperation; alle bits til „0“
NOP 1 <sup>1)</sup>	Disse instruktioner skal indlæses som AWL	Disse instruktioner skal indlæses som AWL		35	Nuloperation; alle bits til „1“
STP	STP	STP		35	Stop; cyklusen bearbejdes færdig og derefter går PLC'en i stoptilstand
BLD	BLD	BLD		35	Billedopbygningskommandoer for programmeringsenheden. Programkørslen i PLC forbliver uændret.

1) Ikke muligt med PG 605U.

Mere detaljeret oversigt findes i manual, afsnit 13.

# Operationsoversigt STEP 5 - Ekstra operationer

Operation	Operandmærke			Bearbejdnings- tid i $\mu$ s	Beskrivelse
	Formal operand	Absolut operand	(tilladt som aktuel operand eller absolut operand)		
kun gyldig i funktions- blokke					
Instruk- tionsliste AWL				CPU 100 CPU 102	
<b>Forbindelsesoperationer</b>					
<b>UW</b>				53	OG-forbindelse på ordniveau (16 bit) af akku 2 med akku 1. Resultatet står i akku 1.
<b>OW</b>				53	ELLER-forbindelse på ordniveau (16 bit) af akku 2 med akku 1. Resultatet står i akku 1.
<b>XOW</b>				51	Eksklusiv-ELLER-forbindelse på ordniveau (16 bit) af akku 2 med akku 1. Resultatet står i akku 1.
<b>Omsætningsoperationer</b>					
<b>KEW</b>				42	Danner ener-komplement i akku 1
<b>KZW</b>				60	Danner toer-komplement i akku 1
<b>Forskydeoperation</b>					
<b>SLW</b>		X	0 til 15 (n)	10n +47	Forskyder indholdet af akku 1 n bit til venstre
<b>SRW</b>		X	0 til 15 (n)	10n +47	Forskyder indholdet af akku 1 n bit til højre

Mere detaljeret oversigt findes i manual, afsnit 13.

# Operationsoversigt STEP 5 · Ekstra operationer

Operation	Operandmærke		Bearbejdnings- tid i $\mu$ s	Beskrivelse
	Formel operand	Absolut operand		
kun gyldig i funktions- blokke		(tilladt som aktuel operand eller absolut operand)	CPU 100 CPU 102	
Instruk- tionsliste AWL				
<b>Springoperationer</b>				
<b>SPA =</b>		X	62	Ubetinget spring
<b>SPB =</b>		X	65	Betinget spring, ved resultat „1“
<b>SPZ =</b>		X	69	Spring ved resultat nul
<b>SPN =</b>		X	69	Spring ved resultat forskelligt fra nul
<b>SPP =</b>		X	69	Spring ved resultat større end nul
<b>SPM =</b>		X	71	Spring ved resultat mindre end nul
<b>SPO =</b>		X	65	Spring ved „overløb“

Symboliste adresse max. 4 tegn

Mere detaljeret oversigt findes i manual, afsnit 13.



# SIMATIC S5-100 U: Tekniske data

<u>CPU</u>	<u>CPU 100</u>	<u>CPU 102</u>	<u>CPU 103</u>
Digitale ind- og udgange	max. 128	256	256
Analoge ind- og udgange	max. 8	16	32
Programlager			
- internt (RAM)	1 k instruktioner	2 k instruktioner	10 k instruktioner
- eksternt (med EPROM- eller EEPROM-lagermodul)	1 k instruktioner	2 k instruktioner	10 k instruktioner
Programbearbejdningstid 1 k instruktioner	70 ms	7 ms	1,6 ms
Merkere	1024 heraf 512 remanente ved indsætning af et-pufferbatteri.	1024	2048
Timere			
- interne (tidsområde 0.01...9990 s)	16	32	128
- eksterne	muligt via timermoduler.		
Tællere			
- interne (tællerområde 0....999; frem/tilbage)	16	32	128
- eksterne	heraf er 8 remanente ved indsætning af et pufferbatteri. muligt via tællermoduler.		
Instruktionsomfang	binære operationer, parenteskommandoer, resultathenvisning, set-reset, tælling, tidsfunktioner, ladefunktioner, transferfunktioner, sammenligning, spring opkald til datablokke, særfunktioner, ordlogik, regning.		
Programmeringssprog	STEP 5		
Programmeringssprog	Instruktionsliste (AWL), funktionsplan (FUP), kontaktplan (KOP).		
Programorganisation	Lineær eller struktureret.		
Strømforsyning	DC 24 V eller AC 115/230 V (via strømforsyningsmodul).		

# SIEMENS

<u>CPU</u>	<u>CPU 100</u>	<u>CPU 102</u>	<u>CPU 103</u>
Digitale indgangsmoduler	4 x DC 24 V, potentialforbundne		
	8 x DC 24 V, potentialforbundne		
	4 x DC 24...60V, potentialforbundne		
	8 x DC 24 V, potentialadskilte		
	4 x AC 230 V, potentialadskilte		
	8 x AC 230 V, potentialadskilte		
Digitale udgangsmoduler	4 x DC 24 V, 0,5 A, potentialforbundne		
	4 x DC 24 V, 2 A, potentialforbundne		
	8 x DC 24 V, 0,5 A, potentialforbundne		
	4 x DC 24...60 V, 0,5 A potentialadskilte		
	8 x DC 24 V, 0,5 A, potentialadskilte		
	4 x AC 115/230 V, 1 A, potentialadskilte		
	8 x AC 115/230 V, 0,5 A, potentialadskilte		
Relæudgangsmoduler	4 x relæ DC 30 V, 2,5 A AC 230 V, 5 A		
Analoge indgangsmoduler	4 x +/- 50 mV, potentialadskilte		
	2 x +/- 500 mV/PT 100, potentialadskilte		
	4 x +/- 1 V, potentialadskilte		
	4 x +/- 10 V, potentialadskilte		
	4 x +/- 20 mA, potentialadskilte		
	4 x + 4...20 mA, potentialadskilte		
Analoge udgangsmoduler	2 x +/- 1..5 V, potentialadskilte		
	2 x +/- 10 V, potentialadskilte		
	2 x +/- 0...20 mA, potentialadskilte		
	2 x 4...20 mA, potentialadskilte		

# SIEMENS

<u>CPU</u>	<u>CPU 100</u>	<u>CPU 102</u>	<u>CPU 103</u>
Programmeringsenheder	PG 605U, PG 615, PG 635, PG 685 og PG 695		
Signalforarbejdende moduler	grænseværdimoduler timermoduler tællermoduler		
Muligheder for bustilslutning	nej	til SINEC L1 (som slave)	til SINEC L (som slave)
Monteringsmuligheder	0...60 grader Celsius		
Relativ fugtighed efter DIN 40040	15%...95% (indendørs), ingen dug		







