

Inledande exempel

HISS-styrning med Mitsubishi A1S PLC

Följande komma-igång-exempel kan ses som ett embryo till lösning av ert PLC-styrproblem inom styrprojektet. Arbeta er igenom detta och sedan skall ni som förberedelse till nästa labtillfälle göra flödesschema och utkast till kod för PLC-styrningen för ert problem.

Följande styrning skall åstadkommas:

Om hisskorg befinner sig på våning 1 och beställning av våning 2 görs antingen från trapphuset eller från hisskorgen så skall lamporna i respektive tryckknapp tändas och hissen gå till våning 2. Då våning 2 nåtts skall hissen stoppa, lamporna släckas och trapphusknapplampor våning 4 tändas och lysa i 5 sekunder. (Observera att detta är ett ofullständigt program. När ni så småningom testat en gång måste styrsystemet resettas och hissen manuellt föras till våning 1 igen.)

Lösning:

Börja med att ge de signaler som skall användas en benämning (tag-namn) enligt:

Insignaler:

Beställning trappa vån2	TRAPP2
Beställning hisskorg vån 2	KORG2
Hiss på våning 1	VAN1
Hiss på våning 2	VAN2

Utsignaler:

Bestlampa trappa vån2	TRLAMP2
Bestlampa hisskorg vån2	HKLAMP2
Bestlampa trappa vån4	TRLAMP4
Hiss uppåt	HISSUPP

Börja programmeringen i GX IEC Developer

Skapa Nytt projekt

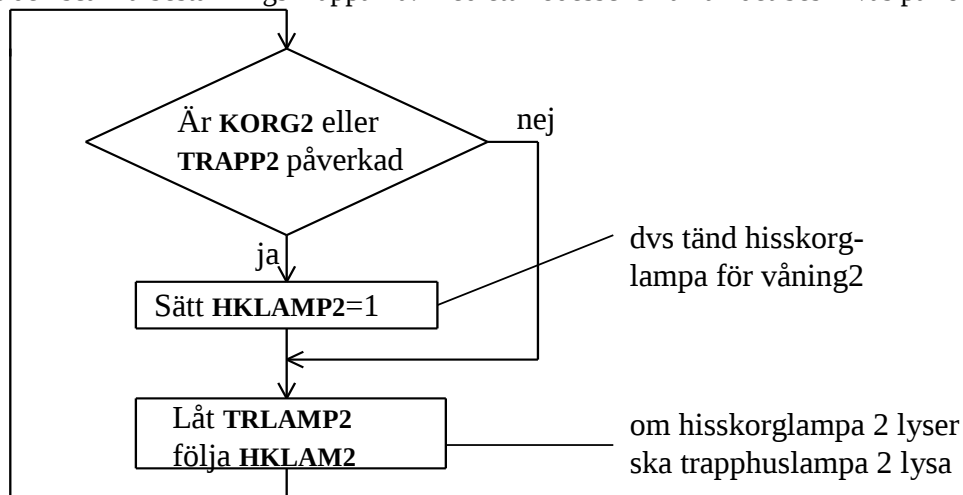
1. Starta GX IEC Developer (under Automation – Melsoft -)
2. Välj **Projekt > New**
3. Välj CPU Type: **AnS A1S**
4. I fönstret New Project skrivs sökväg och namn på katalogen där filerna i projektet ska hamna: (katalognamn=projektnamn). OBS! max 8 tecken och inte åäö mm.
Skapa projektet på din egna area på. Ex. Z:\styrtekn\projnamn. Klicka **Create**
5. I dialogrutan New Project Startup Option välj: **Empty Project**.
Om en dialogruta "Library update check" dyker upp klicka Close

Deklarera Globala variabler

6. I fönstret Project Navigator, dubbelklicka på **Global_Vars**
Röd *Asterisk före någon rubrik betyder att enheten inte är kompilerad.
7. Lägg in din variabeldeklarationslista med Identifier och MIT-Addr.
(Motsvarande IEC-standardadress fås automatiskt och typen blir automatiskt BOOL)
Alla variabler som är kopplade till en in- eller utgångsadress måste vara globala.
Observera att GX IEC Developer är "case insensitive" d v s programmet bryr sig inte om variabler är skrivna med versaler eller gemener.
8. Ny rad fås med 3:e verktygsknappen från höger. 
9. Då globala variabellistan är klar ska den se ut enligt nedan. Tillordning dvs hur de olika signalerna är kopplade till PLC-systemet framgår av tillordningslista i "Projektoppgiftshäftet".
OBS. Stäng fönstret Global Variable List för att ändringar ska utföras

	Class	Au	Identifier	MIT-Addr.	IEC-Addr.	Type
0	VAR_GLOBAL	↓	TRAPP2	X1	%IX1	BOOL
1	VAR_GLOBAL	↓	KORG2	X9	%IX9	BOOL
2	VAR_GLOBAL	↓	VAN1	X4	%IX4	BOOL
3	VAR_GLOBAL	↓	VAN2	X5	%IX5	BOOL
4	VAR_GLOBAL	↓	TRLAMP2	Y25	%QX37	BOOL
5	VAR_GLOBAL	↓	TRLAMP4	Y27	%QX39	BOOL
6	VAR_GLOBAL	↓	HKLAMP2	Y21	%QX33	BOOL
7	VAR_GLOBAL	↓	HISUPP	Y28	%QX40	BOOL

Beställningsknapparna kring hissen måste iakttas hela tiden av PLC-systemet dvs styrsystemet måste hela tiden scanna beställningsknapparna. Med ett flödesschema kan det beskrivas på följande sätt:



Det är dags att skapa ett POU (delprogram) som uppfyller funktionen som beskrivs i flödesdiagrammet.

Skapa POU

10. I fönstret Project Navigator markera **POU Pool**

Klicka på verktygsknapp **POU** (4:e från höger)



Namnge din POU.

Välj Class: **PRG**

Välj Language of the Body: **Function Block Diagram**

Klicka **OK**

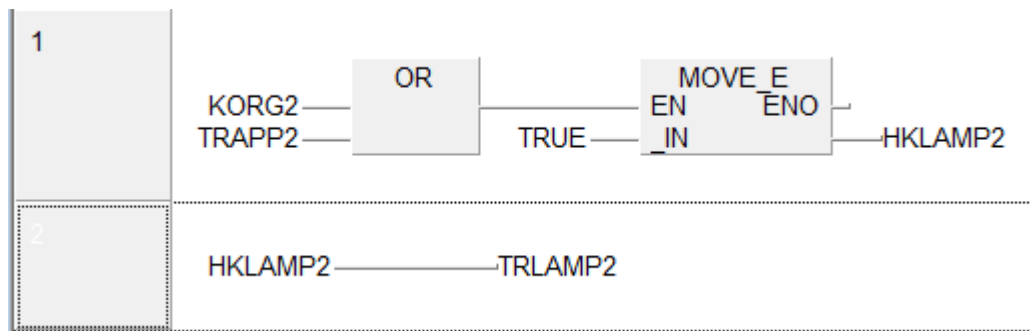
(Svara **Yes** i eventuell dialogruta som dyker upp)

11. I Project Navigator, dubbelklicka på ditt **POU-namn**.

Dubbelklicka på **Header** för att få fram det som är listan för till detta POU lokala variabler. Inga lokala variabler behöver tillföras listan nu men kan senare kompletteras.

12. I Project Navigator, dubbelklicka på **Body**. En Networkruta finns nu i Body fönstret.

Nedanstående Body byggd av funktionsblock (FBD) skall skapas som motsvarar det tidigare givna flödesschemat.



För att välja funktionsblock: Klicka på "IC-kapselsymbolen" i verktygsraden.



Dialogruta Function Block Selection öppnas. Under Operator Type, välj **All Types**

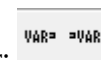
Hämta **OR**-blocket, klicka **APPLY** och placera funktionsblocket i Networkrutan.

Leta på motsvarande sätt, fram **MOVE_E**-blocket och placera det i Networkrutan.

För beskrivning av de olika funktionsblocken markera funktionsblocket i Dialogruta Function Block Selection och tryck Help.

Om funktionsblocket inte har "?" på in- och utgångar för inskrivning av in- och utvariabler så är inte Auto Connect aktiverad. Högerklicka i så fall i en Networkruta, välj Auto Connect och hämta blocket på nytt.

Om man vill addera "?" på in- och utgångar i efterhand välj respektive knappar:



Om man vill invertera en in- eller utgång, dubbelklicka alldeles innanför anslutningen till blocket.

13. Skapa ny Networkruta genom att klicka på Verktygsknapp (samma knapp som för nya rader i Globala variabellistan)



Lägg till en insignal och en utsignal i denna ruta med knapparna



14. Stäng Function Block Selection.

Klicka på en in- eller utgång "?" på ett funktionsblock för att lägga in variabelnamn på något av följande sätt:

- Skriv in din variabel ("Identifier") direkt.
- Högerklicka på "?" för att få en variabellista att välja från. Välj önskad variabel klicka **Apply**

15. Återstår att förbinda utgång på OR-krets med ingång på MOVE_E-kretsen. Högerklicka i Networkruta.

Välj **Interconnect Mode**. Eller välj förbindelseknappen i menyn:



Klicka på utgång och dra en förbindelse till ingång, klicka på ingång. Upprepa för förbindelsen mellan HKLAMP2 och TRLAMP2.

Då alla förbindelser klara Högerklicka igen, välj **Select Mode**.

Alternativt klicka på pilsymbolen i menyn:

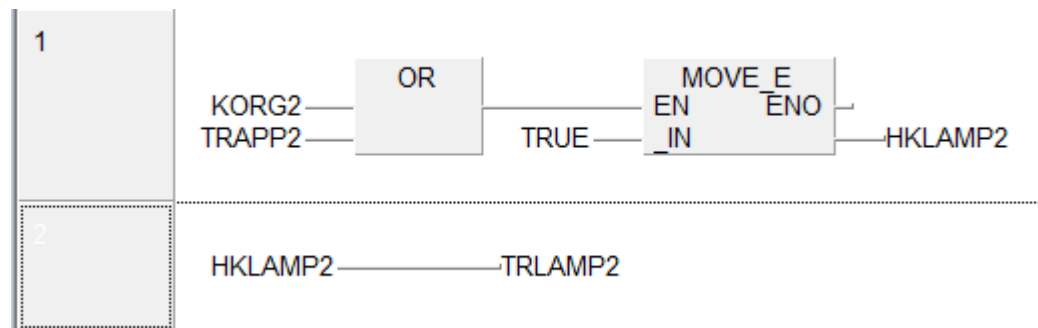


Om man inte kan dra förbindelser så är inte Auto Connect aktiverad. Högerklicka i så fall i en Networkruta, välj Auto Connect.

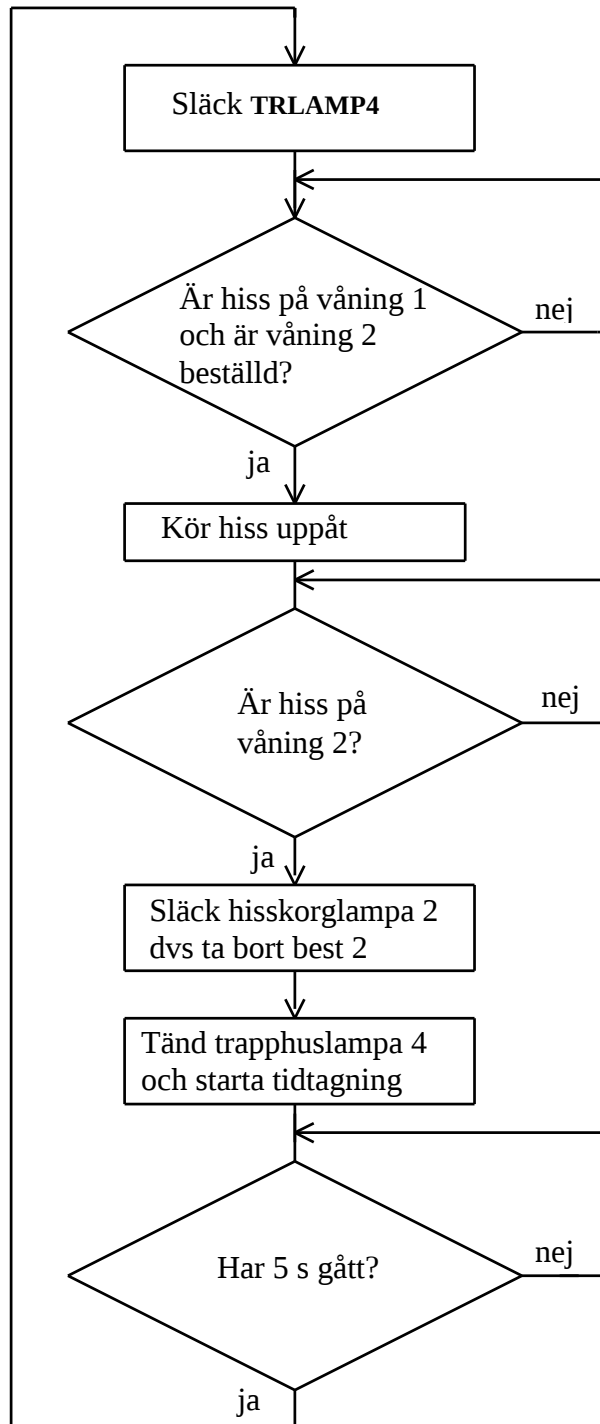
Infoga kommentarer:



Body i din POU borde nu se ut som figur nedan.

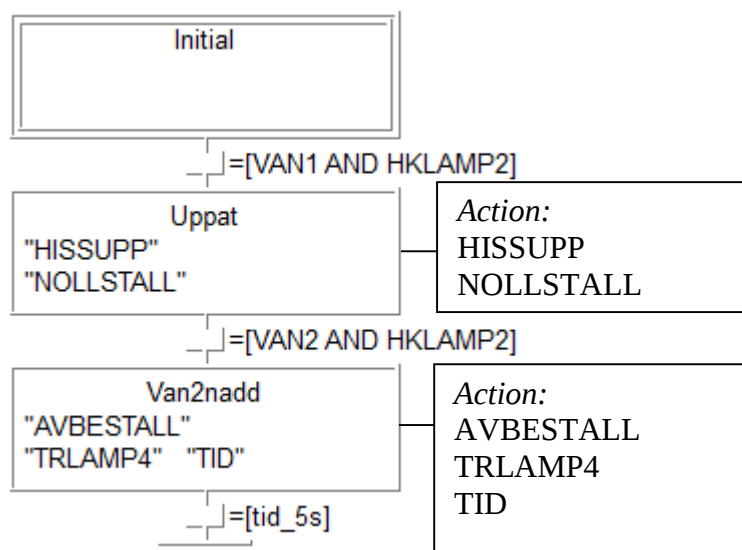


Själva styrsekvensen för hissen innebär att den väntar på beställning och då beställning kommit kör den uppåt ända tills den nått fram då tar nästa sak vid osv. Med ett flödesschema kan det beskrivas på följande sätt:



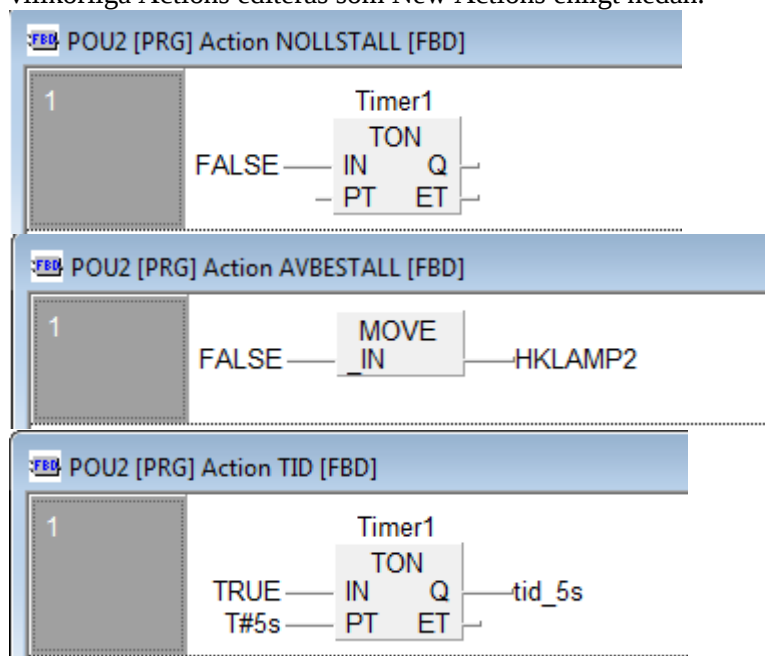
Skapa ett POU som ges namnet Hissekvens och välj programmeringssätt Sequential Function Chart (SFC) som är gjort för att använda för sekventiella förlopp.

Nedan framgår SFC-koden för ovanstående flödesdiagram. Följ beskrivningen från punkt 16 för hur detta byggs upp.



Obs! Action-rutorna i figuren här intill syns inte på detta sätt i GX IEC – editorn.

I steg Uppat aktiveras ovillkorligt HISSUPP. I steg Van2nadd aktiveras ovillkorligt TRLAMP4. Övriga villkorliga Actions editeras som New Actions enligt nedan.



Actions AVBESTALL och TID i steg Van2nadd samt NOLLSTALL i steg Uppat ser ut enligt ovan. AVBESTALL tar bort beställningen dvs. reset av HKLAMP2 vilken sätts i det tidigare behandlade FBD-POU:et för våningsbeställning. Det innebär att trapplampan TRLAMP2 också släcks då den slaviskt följer HKLAMP2. Hantering av HKLAMP2 i de två olika POU:erna förutsätter att HKLAMP2 är deklarerad som Global Variabel, vilket den ju är.

I action TID aktiveras en timer av typ TON. Trots att EN-ingången sätts TRUE innebär det här att timern inte börjar ta tid förrän man är i steget Van2nadd. När 5 sekunder gått är övergångsvillkoret tid_5s under detta steg uppfyllt och övergång sker till steg Initial. Notera att ni måste nollställa timern igen innan ni låter den börja räkna nästa gång, annars kommer tid_5s att vara TRUE direkt när ni kommer in i Van2nadd. Denna nollställning görs genom att samma timer (notera att de har samma namn, Timer1) ges FALSE som insignal i steg Uppat. Nollställningen skulle också kunna göras i ett steg som ligger efter Van2nadd. Det viktiga är att det inte görs i samma steg.

Åter till programmeringen - skapa ett SFC-POU

16. I fönstret Project Navigator markera **POU Pool**

Klicka på verktygsknapp **POU** (4:e från höger)



Namnge din POU (Hissekvens).

Välj Language of the Body: **Sequential Function Chart**

Klicka **OK**

(Svara **Yes** i eventuell dialogruta som dyker upp)

Ändra standardinställningar i GX IEC Developer så att bl a kommentarer i SFC syns:

View>Extended Information>SFC editor

17. I Project Navigator, dubbelklicka på **Body**

En grundstomme till ett SFC-program finns nu i Body fönstret.

Bygg upp strukturen på ditt program enligt nedan:

Infoga nya steg och övergångar med verktygsknapp:

(Infoga parallella/alternativa processer med verktygsknappar:

(Infoga hopp med verktygsknappar:

(Infoga endast steg med: Infoga endast övergång med:

18. Det sista steget (Final Step) visas som ett tvärstreck och ger återhopp till Initial Step.

Om du råkar radera: Gå till **Edit > Modify > Final Step**

19. Klicka på dina steg och namnge varje steg med ett unikt namn.

20. Klicka på övergångar och ange övergångsvillkor på något av sätten nedan:

- Om övergångsvillkor är enkelt: En variabel: eller en enkel funktion av variabler:

Tools > Edit Transition Condition och skriv in funktionen: T ex: **VAN1 AND HKLAMP2**

Vid övergången syns nu: = [övergångsvillkor]

Fyll på detta sätt i alla övergångsvillkor.

Om övergångsvillkor är komplicerat: Skriv in namnet på övergången

Dubbelklicka på övergång. Välj Language: T ex **Function Block Diagram**.

Programmera som FBD enligt tidigare och benämnen den enda resulterande utgången med TRAN.

TRAN ettställd innebär att övergången sker.

21. För att lägga in vad som skall hända i varje steg dubbelklicka på steget och ange de händelser (aktiviteter) som ska vara associerade till steget. Olika händelser på olika rader i Action Association fönstret.

Händelserna kan vara av 2 slag:

- En variabel, vilket är fallet i detta exempel när det gäller aktivering av HISSUPP. (*Variabeln är 1 (TRUE) endast under den tid steget är aktivt*)
- En händelse i form av ett program s k Action:
Skriv in namnet på din Action i Action Association-fönstret



Klicka på verktygsknapp

Välj programspråk. Klicka **OK**

Klicka på Action namn i Project Navigator.

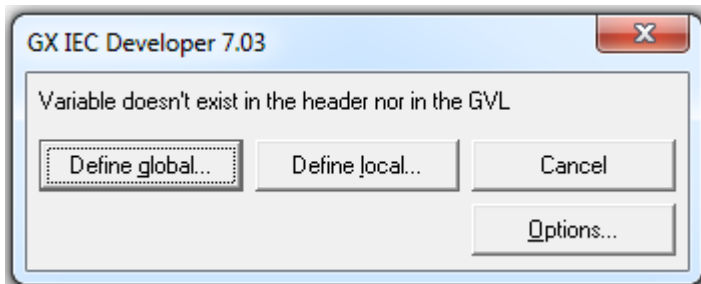
Programmera dina händelser (aktiviteter) enligt figurena ovan.

Några benämningar kommer att införas som inte finns i Globala Variabellistan enligt ovan. .
Flaggan "tid_5s" är av typ BOOL och används bara internt i detta POU varför den deklarerats som lokal variabel i Headern.

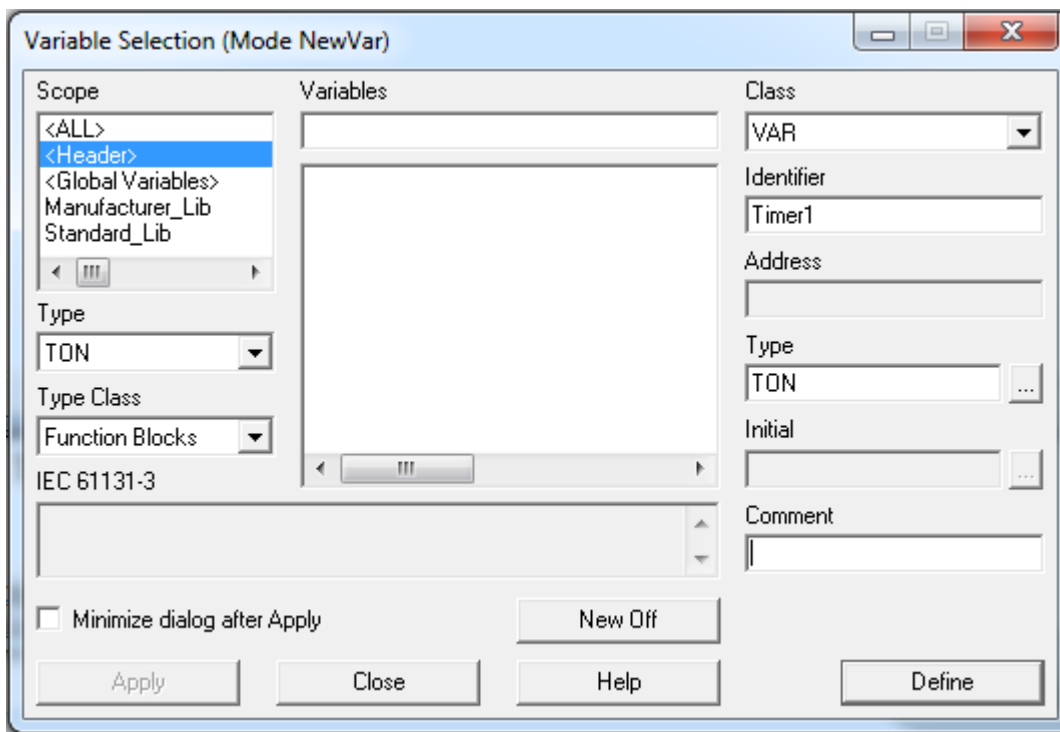
Instance "Timer1" är en annan ny benämning. TON-blocken är en typ av Function Block som innehåller adressering till något intern minnesflagga, register, räknare och/eller timer. Blocken måste därför vara unika dvs ha ett unikt Instance-namn som registreras i Headern. För att ange unikt Instance namn:

Klicka på "Instance". Skriv in ett unikt namn. Tryck Enter

I dialogruta som dyker upp enligt nedan: Välj: **Define local**



I ny dialogruta som dyker upp enligt nedan:



Välj: **Define**

För att infoga kommentarer för varje steg. Klicka på steget och välj kommentarknapp
Skriv dina kommentarer inom citationstecken för att skilja dem från stegets namn.
Det kan vara lämpligt att ange alla actions i steget i kommentarer.



Skapa Task

23. I Project Navigator markera **Task_Pool**

Klicka på verktygsknapp **TSK**
Namnge din Task.



24. I Project Navigator dubbelklicka på ditt **Task-namn**

I ditt TASK-fönster ska du välja vilka POU som ska ingå. (I detta fall finns två POU:er. Lägg till rad på redan känt sätt.) Skriv in ditt **POU-namn** (eller välj från lista m h a knapp:

	POU name		Comment
0			

Kompilering av projekt

25. Kompilera projektet

Välj **Project > Build**



Röd *Asterisk före någon rubrik betyder att enheten inte är kompilerad.
Beror oftast på att den aktuella POU:n ej är kopplad till någon Task.

26. Om meddelandet: The program is too big for actual configuration dyker upp, gör följande: I Project Navigation: Dubbelklicka på PLC
Ändra: Program capacity > Microcomputer (0-14) till: **2 Kbytes**
27. Förhoppningsvis inga fel? Korrigera annars. Genom att markera felet i kompileringslistan och trycka "Show" leds man till rödmarkerade felstället.

Överföring av program till PLC

28. Om du inte har kopplat ihop PLC:n med processen du ska styra så gör det nu innan du ansluter PLC:n till matningsspänning.
- 29 För överföring av det kompilerade programmet till PLC

Project > Transfer > Download to PLC. 

Vid överföring måste programexekveringen i PLC:t stoppas vilket görs genom att svara på fråga i början på programöverföringen. Lyckas inte överföringen gå till:

Online > Transfer Setup > Ports.

Klicka Serial och välj RS-232C, COM1. Programmet ligger kvar i PLC:s programminne tills ett nytt program överförs. Reset av PLC innebär nollställning av flaggor och register men inte radering av program. (Om PLC:n begär Keyword, Skriv: **fffff**.) Starta exekvering genom att svara på fråga då överföring klar. Run/Stop/reset-hantering finns under Online-meny.

ON-line-ändring.

30. Mindre ändringar av av programmet kan göras on-line med PLC i RUN-läge. Gör ändringarna i GX IEC, kompilera inte utan välj **Project > Online Program Change**

Monitorering vid körning

31. Starta exekvering genom att vrida nyckel på PLC-CPU till RUN. Gå till Online och aktivera

Monitoring Mode.  Signaler och tillstånd kan nu följas i de olika POU:erna. Klicka Start Monitoring under Online för varje POU.

Spara Projektet på egen area.

- 32 Vid några tillfällen har vi observerat instabilitet och förlorade projekt då man jobbat över nätet mot Z: Om du har problem med detta jobba mot egna lokala enheten C: på labbdatorernas hårddisk. Kom dock ihåg att spara på er egen Z: -area innan ni loggar ut. Det kan göras genom att gå till Extras – Project Backup och spara det i en backup-fil av typ *.pcd. Projektet kan sedan öppnas igen via Extras – Project Restore. Ett annat sätt är att använda Save As på vanligt sätt men då blir projektet ganska utrymmeskrävande. (En tredje variant är att markera Project i Navigatorn och välja Export. Då komprimeras projektet till en .asc – fil som kan importeras på motsvarande sätt nästa arbetstillfälle.)
Obs! Ta för vana att då och då under ett labpass att göra en backup-sparande ifall något skulle hända när ni jobbar med programmet.