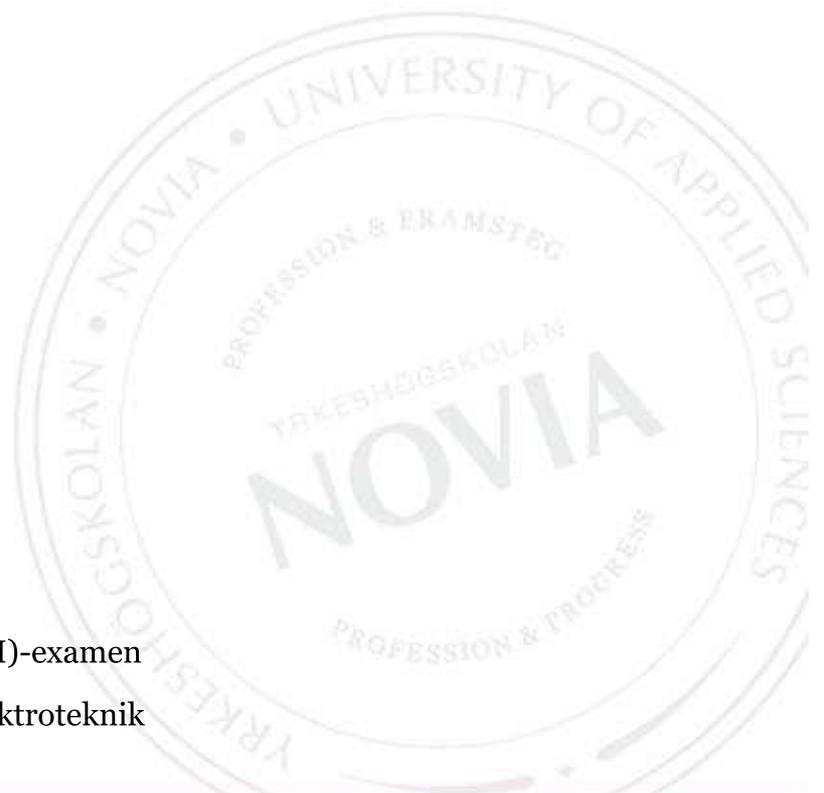




# **PROCESSMODELL**

## **MODERNISERING OCH UPPDATERING AV PROCESSMODELL OCH DESS STYRSYSTEM**

KJELL SKYTTE



Examensarbete för ingenjör(YH)-examen

Utbildningsprogrammet för elektroteknik

Vasa 2011

## EXAMENSARBETE

Författare: Kjell Skytte  
Utbildningsprogram och ort: Elektroteknik, Vasa  
Inriktningsalternativ: Automationsteknik  
Handledare: Erik Englund

Titel: *Modernisering och uppdatering av processmodell och dess styrsystem*

---

1.4 2011                      21 sidor

---

### **Abstrakt**

Examensarbetet omfattar en modernisering och en uppdatering av en processmodell för undervisningsbruk. Processmodellen är ett resultat av ett tidigare ingenjörsarbete, konstruktion av processundervisningsapparat, utfört av Robert Pernu 1989-90. Den Allan-Bradley logik som installerades för 20 år har tjänat ut sitt syfte och ersattes med ny Siemens logik. Huvudmomenten i examensarbetet är, kartläggningen av den befintliga anläggningen, planeringen av den ersättande logiken, programmeringen av densamma samt tillhörande dokumentation. Målet att få en uppdaterad och fungerande processmodell har uppnåtts.

---

Språk: svenska

Nyckelord: processmodell, PLC, sensorer

---

Förvaras: Examensarbetet finns tillgängligt antingen i webbiblioteket Theseus.fi eller i Tritonia, Vasa vetenskapliga bibliotek.

## BACHELOR'S THESIS

Author: Kjell Skytte  
Degree programme: Electrical Technology  
Specialization: Automation  
Supervisor: Erik Englund

Titel: *Modernizing and updating a process model and the control system of the model*

---

1 April 2011

21 pages

---

### **Abstract**

This thesis work consists of modernizing and updating a process model to be used in education. The process model is a result of an earlier thesis made by Robert Pernu during the academic year 1989-90. Its name was "Construction of a process model for use in education".

The Allan-Bradley logic installed 20 years ago had become outdated and has been replaced by a new Siemens logic. The main contents of the thesis are mapping of the model, planning the replacing logic, programming of the logic, and documentation. The goal to get an updated and working process model has been achieved.

---

Language: Swedish

Key words: process modell, PLC, sensors

---

Filed at: The Bachelor's thesis is available either at the web library Theseus.fi or at Tritonia Academic Library, Vaasa.

# INNEHÅLLSFÖRTECKNING

**ABSTRAKT**

**ABSTRACT**

**INNEHÅLLSFÖRTECKNING**

## 1 Inledning

1.1 Uppdragsgivare.....	1
1.2 Projektet.....	1
1.3 Processundervisningsmodellen.....	2
1.4 Uppdateringsuppgiften.....	3

## 2. PLC-styrningen.....4

2.1 Maskinvara och arkitektur.....	4
2.2 Logik.....	5
2.3 Programvara.....	6

## 3 Funktionella krav på processmodellen.....7

3.1 Processmodellens in- och utgångar.....	7
3.2 Processekvenser.....	10

## 4 Specifikation av ett nytt styrsystem.....12

4.1 Simulation.....	12
4.2 Relästyrningar.....	14
4.3 Styrprogrammet.....	14

## 5 Ombyggnad.....15

5.1 Konstruktion.....	15
5.2 Modellerings.....	16
5.3 Programmering.....	17

## 6 Resultat.....19

## 7 Utvärdering och diskussion.....19

## 8 Källförteckning.....21

# MODERNISERING OCH UPPDATERTING AV PROCESSMODELL OCH DESS STYRSYSTEM

## 1 Inledning

I detta dokument förklaras hur och på vilket sätt de olika delmomenten i uppdateringsprocessen kommer att gestalta sig. Hela uppdateringen och medföljande dokumentation är en arbetsdryg process. Den programkod i den nya logiken som kommer att motsvara den gamla, är en plattform för vidare utveckling.

### 1.1 Uppdragsgivare

Yrkesakademin i Österbotten, tidigare känt som Svenska yrkesinstitutet (SYI), el-avdelningen har verkat som uppdragsgivare och erbjöd mig möjligheten att modernisera och uppdatera deras unika processundervisningsapparat.

### 1.2 Projektet

Som det tidigare nämnts så handlar detta lärdomsprov om att modernisera och uppdatera den befintliga styrutrustningen på processmodellen. Med tiden så blir den styrlogik som för stunden är modern, ohjälpligt föråldrad. Speciellt gäller detta programvara och styrsystem. Den Allan-Bradley logik som för 20 år sedan med dåtida mått var det bästa marknaden hade att erbjuda, har nu fått musiestatus. Eftersom processmodellen är till sin konstruktion helt unik, så var ett av kraven att dess styrlogik måste förnyas för att den igen skall kunna tjäna i undervisningssyfte.

### 1.3 Processundervisningsmodellen

Processmodellen hänför sig till en lärokurs i industri installationer, vars mål var bl a. att ge studeranden grundläggande färdigheter i eldrifternas styr- och driftsapparater, elritningar och installation av styr- och kraftkablar. Provning av en färdig installation och felsökning i den samt därtill installation, användning och programmering av programmerbar logik ingick också i kursen.



Figur 1. Processmodellen

För att visa hur en PLC används, hur man löser och programmerar, för att sedan ta idrift ett PLC-system, så är en verklig modell ett utmärkt verktyg för ändamålet. Den erbjuder motsvarande möjligheter som ett riktigt system ute i industrin. Det ger möjligheter att korrigera programfel och undersöka olika lösningar, utan att det kan ske någon större katastrof, vilket inte är fallet i verklig processkörning.

Den unika processundervisningsmodellen som finns på YA:s enhet i Brändö, är en konstruktion som tillkommit genom ett ingenjörarbete på VTLA åren 1989-90. Apparaturen består av i en elevator, en kedjetransportör, en remtransportör samt två stycken skruvtransportörer. Denna anläggning anskaffade man ha till yrkesskolan för laborationsändamål. I dessa transportörer kan man köra plastgranulat och körningen kan styras med programbar logik. I anläggningen finns även fyra stycken behållare samt två stycken grenrör med diverse spjäll. Spjällen, totalt tre stycken, fungerar med pneumatiska tryckluftscylindrar samt en spindelmotor. Maskineriet får sin styrning från en motorcentral och kan fjärrmanövreras från en manövercentral.

Det befintliga Allan-Bradley styrsystemet var försett med reläutgångar, medan ingångarna var digitala. Styrspänningen +24 V, användes både som insignal från givarna och som styrspänning via reläutgångarna till motorernas kontaktorer. Styrutrustningen bestod av en processorenhet samt två expansionsenheter, som även matades med samma +24 V.

#### 1.4 Uppdateringsuppgiften

Till processmodellen finns kopplad en motorcentral med tillhörande manövercentral. I och med att logikstyrningen och dess programmering föråldrades, kom anläggningen med tiden att bli oanvändbar. Därför var det en nödvändighet att uppdatera dess styrning till en modernare variant, för att studeranden skulle kunna utföra relevanta uppgifter. Själva anläggningen var helt intakt, varför min uppgift blev att uppdatera dess logikstyrning.

## 2. PLC-styrning

Det var i slutet av 1950-talet som man inom bilindustrin i USA började utveckla nya styrsystem. Orsaken var, att det blev kostsamt och tidsödande att bygga om de krävande styrsystemen vid byte av bilmodeller. Den tekniska utvecklingen har gått vidare till den vi har i dag, i form av industrirobotar med avancerade styrsystem. Oavsett om det gäller enkel mekanisering eller mer avancerad automatisering, bygger utvecklingen på ett samspel, en förståelse mellan människa och maskin.

I takt med att ny teknik införs och industrin utvecklats, uppkommer det ständigt nya möjligheter att mekanisera, styra och övervaka processer genom automatisering. Då allt högre krav ställdes på flexibilitet och effektivitet uppkom det vi kallar PLC-tekniken. PLC står för "Programmable Logical Controller" dvs. Programmerbart Logiskt Styrssystem. Att PLC har blivit grunden i moderna styrtekniska lösningar, beror på dess flexibilitet. Man kan enkelt anpassa styrsystemet efter just de arbetsuppgifter som behövs, genom omprogrammering. (Kördel Lennart & Bengtsson Peter 1998, s. 26 och s. 54)

### 2.1 Maskinvara och arkitektur

En PLC är i princip uppbyggd av en centralenhet, CPU-enhet, samt ingångs- och utgångsenheter. För programmering används serie- eller usb-kablage, kopplat till PC :n tillsammans med programmeringsmjukvara.

För att kunna erhålla information till PLC-enheten behövs mätton som kallas sensorer eller givare. Då givaren påverkas av processen ger den en utsignal som indikerar en förändring av tillståndet hos nämnda givare, t ex. från logisk "0" till logisk "1". Givare finns för många olika ändamål och tillämpningar. De kan ge information om gränsläge, magnetfält, induktivt- och kapacitivt kraftfält, samt olika slag av ljus. I givarna finns monterade mekaniska kontakter eller halvlederledarkomponenter. De mekaniska kontakterna kan vara av öppnande eller slutande typ medan halvledarmodellerna har en PNP- eller NPN-funktion som utsignal.

## 2.2 Logik

Grunden för PLC-programmering är de logiska grundfunktionerna. Det finns tre grundläggande funktioner inom styrtekniken. Dessa är OCH- ELLER-, samt INTE-funktionerna, eller på engelska AND-, OR- och NOT-funktionerna. Dessa anger beroendet mellan in- och utgångarna på styrlogiken. Vanligen låter man storhetens största värde representera ett 1-tillstånd (logisk "1") och det lägsta värdet ett 0-tillstånd (logisk "0"). Om man bygger upp ett system enligt föregående beskrivning erhåller man det som kallas positiv logik/hög representation, även det motsatt kan förekomma. Genom att kombinera de olika grundfunktionerna erhåller man lösningar för att klara av de mest varierande styruppgifter, som även kallas kombinatorik.

Om den logiska kretsens uttillstånd entydigt bestäms av rådande logiska tillstånd hos ingångarna så är kretsen kombinatorisk.

Genom att komplettera en ELLER-element med en INTE-element på utgången bygger man en INTE-ELLER funktion, som på engelska kallas NOT-OR eller NOR.

Genom att komplettera en OCH-element med en INTE-element på utgången bygger man en INTE-OCH funktion, som på engelska kallas NOT-AND eller NAND.

Genom att komplettera ett ELLER-element med två OCH-element och ett INTE-element så bygger man upp en EXkluderande ELLER funktion, som på engelska kallas Exclusive-OR eller XOR. Funktionens villkor är att utgången intar 1-tillstånd om en, och endast en, av ingångarna har 1-tillstånd. Sätter man en INTE-element på XOR, så får man en XNOR, ett ska. IDENTITY-element (=likhets-element). Funktionens villkor är att utgången intar 1-tillstånd, endast om båda ingångarna har 0- eller 1-tillstånd.

Förutom dessa så behövs ställ- eller återställfunktioner som vanligen kallas minnen. Dessa finns om A-dominanta (SetReset, SR-krets) eller B-dominanta (RS-krets).

Även funktioner som fördröjer signalen så att den förs vidare först efter en viss tid, kan behövas. Dessa fördröjningsfunktioner används som fördröjt tillslag och fördröjt frånslag, som på engelska kallas ON- eller OFF DELAY.

Dessutom kan man behöva använda sig av en funktion som räknar inkommande pulser och ger signal efter att ett visst antal pulser har uppnåtts: Dessa räkarnarfunktioner används som upp- eller ner räknare, som på engelska kallas UP- eller DOWN-COUNTER.

Ställs man inför en kompliserad styrteknisk uppgift med många ingångar som är beroende av varandra, så kan användandet av Boolesk algebra (=kopplingsalgebra) vara ett sätt att lösa uppgiften, genom att förenkla och skapa ett blockshema, ett s.k. MAJORITY-element. (Kördel Lennart & Bengtsson Peter 1998, s. 28-52)

## 2.3 Programvara

Det finns två typer av funktionsprinciper för en PLC-enhet, cyklisk- och realtidsbaserad exekvering. Den cykliska exekveringen är kontinuerlig och tidsbaserad medan den realtidsbaserade är deterministisk, prioritetbaserad och kan handha multitasking funktioner.

Det finns även ett antal programmeringsspråk att välja mellan då man programmerar. För att erhålla en viss standardisering inom programmering så har man utvecklat en standard innehållande åtta delstandarder, gemensamt kallad IEC 61131. Tanken att man har leverantöroberoende programvaruelement har bidragit till standardens uppkomst, dock så har verkligheten visat att ”tanken om leverantörsoberonde programvara” fortfarande är kvar i sin linda.

Standarden definierar fem programmeringsspråk, av vilka två är textbaserade och tre är grafiska. Till de grafiska språket hör det programmeringsspråk som jag använder, programmering med FBD-block.

### 3 Funktionella krav på processmodellen

Uppgiften var att förnya processmodellens styrsystem, att byta ut den föråldrade A&B till en modell med nyare programmeringsteknik, dock med bibehållen funktionalitet. Eftersom YA sedan länge har använt sig av Siemens Logo i sin automationsundervisning, så var det ett naturligt val att välja Siemens S7-200 för att uppdaterade styrutrustningen. Ett alternativ hade också varit att välja den följande kraftigare modellen av S7-serien, modell 300. Men med tanke på att steget mellan Logo och den förnyade styrutrustningen skall vara möjligast smidig, var det motiverat att använda sig av mellanmodellen S7-200 för detta ändamål.

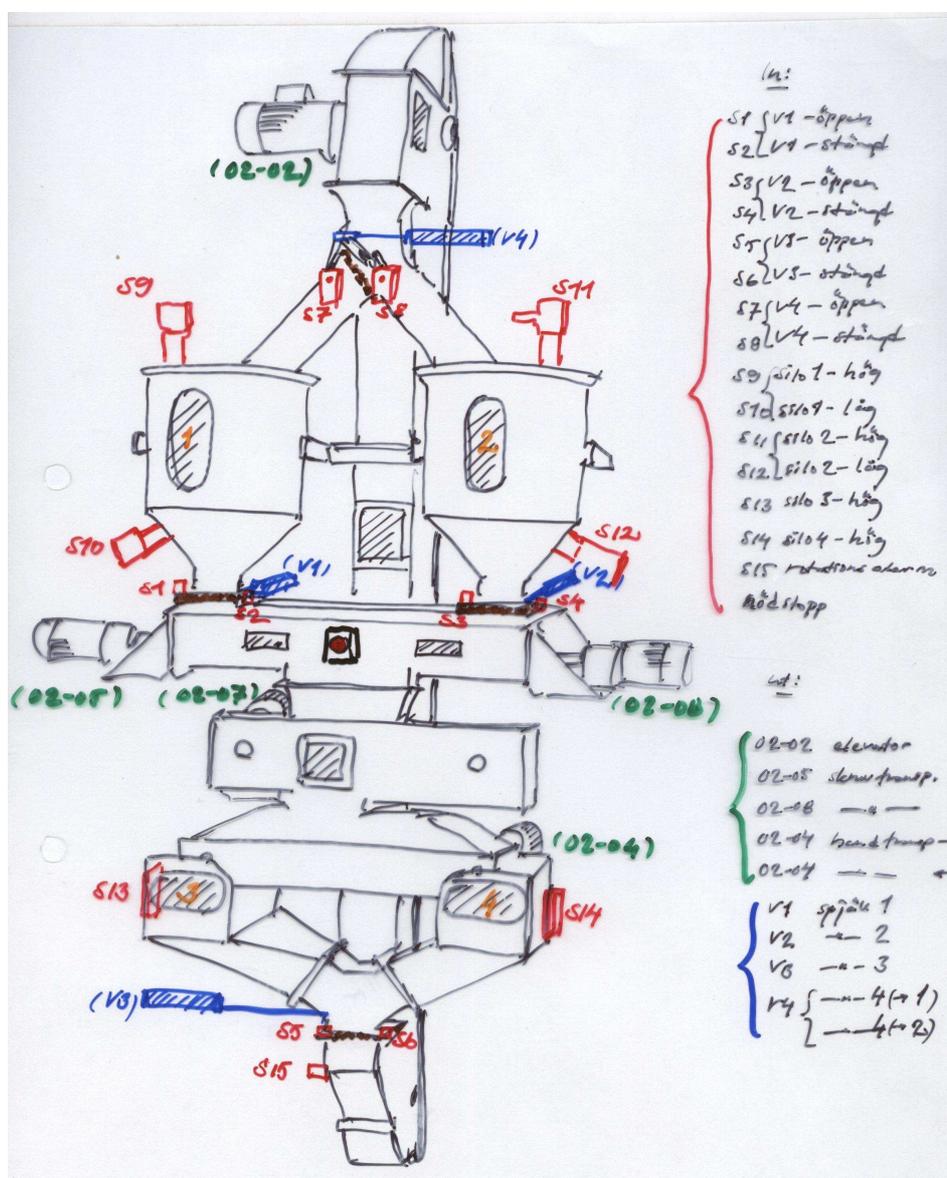
#### 3.1 Processmodellens in- och utgångar

Ingångarna är inkopplade för styrning med positiv logik, ( låg insignal = 0). För givare med halvledarkomponenter betyder det PNP- modell. För att få information om hög- och låg nivå i silon så används justerbara kapacitiva givare. Eftersom PLC-enheten har transistorstyrda utgångar kopplas relä in för vare utgång för att ge styrsignaler till motorer och ventiler.

<b>I/O- lista:</b>	<b>A&amp;B SLC100</b>	<b>Siemens S7-200</b>
Program(val) 1	A001	I0.0
Program(val) 2	A002	I0.1
Program(val) 3	A003	I0.2
Program(val) 4	A004	I0.3
Start(knapp)	A005	I0.4
Stop(knapp)	A006	I0.5
Rotationsalarm	A105	I0.6
		I0.7 (option: elevatorgivare)
Autokörning "02-04"	A007	I1.0
Autokörning "02-08"	A008	I1.1
Autokörning "02-05"	A009	I1.2
Autokörning "02-07"	A010	I1.3

<b>I/O- lista:</b>	<b>A&amp;B SLC100</b>	<b>Siemens S7-200</b>
Givare Spjäll 1 öppet (S1)	A101	I1.4
Givare Spjäll 1 stängd (S2)	A102	I1.5
Givare Spjäll 2 öppet (S3)	A103	I1.6
Givare Spjäll 2 stängd (S4)	A104	I1.7
Givare Spjäll 3 öppet (S5)	A207	I2.0
Givare Spjäll 3 stängd (S6)	A208	I2.1
Givare Spjäll 4 öppet (S7)	A209	I2.2
Givare Spjäll 4 stängd (S8)	A210	I2.3
Givare Silo 1 hög (S9)	A201	I2.4
Givare Silo 1 låg (S10)	A202	I2.5
Givare Silo 2 hög (S11)	A203	I2.6
Givare Silo 2 låg (S12)	A204	I2.7
Givare Silo 3 hög (S13)	A205	I3.0
Givare Silo 4 låg (S14)	A206	I3.1
Styrpanel led "02-07"	A106	I3.2
Styrpanel led "02-05"	A107	I3.3
Styrpanel led "02-08"	A108	I3.4
Styrpanel led "02-04"	A109	I3.5
Styrpanel led "02-04"	A110	I3.6
		I3.7 -
Kedjetransportör "02-07"	A011	Q0.0
Skruvtransportör "02-05"	A012	Q0.1
Skruvtransportör "02-08"	A013	Q0.2
Remtransportör "02-04"->silo 3	A014	Q0.3
Remtransportör "02-04"->silo 4	A015	Q0.4
		Q0.5 (option: elevator "02-02")
Alarm, led H1	A113	Q1.0
Körning OK, led H2	A114	Q1.1
Rotationsgivare OK, led H3	A115	Q1.2
(tomma A016,A111, A112 och A116)		Q1.3-Q1.7 tomma

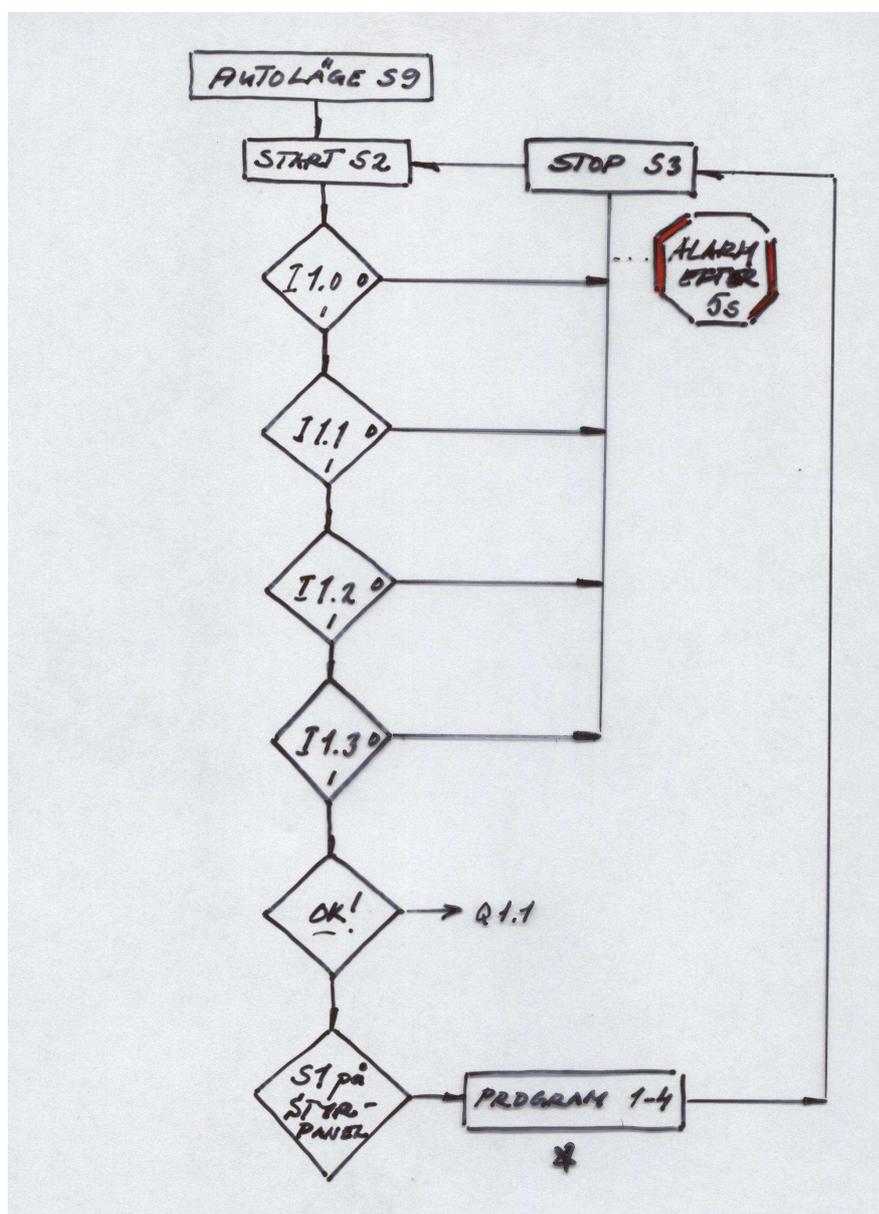
I/O- lista:	A&B SLC100	Siemens S7-200
Styrning magnetventil V1	A211	Q2.0
Styrning magnetventil V2	A212	Q2.1
Styrning magnetventil V3	A213	Q2.2
Styrning magnetventil V4	A214	Q2.3
(tomma A115 och A116)		Q2.4-Q3.7 tomma



Figur 2. Skiss på givare och styrdon

### 3.2 Processekvenser

Processmodellen kan köras både manuellt och med automatik. Möjligheten till manuell körning är viktig i felsituationer som kan uppstå vid en programkörning. De sekvenser som man kört är fyra till antalet. En programväljare med programval 1- 4, varmed man kan välja på vilket sätt man önskar köra från övre till nedre silo. Granulatet körs sedan manuellt till valbar övre silo inför följande valbara automatiska programkörning.



Figur 3. Flödesskema skiss. Inititering



## 4 Specifikation av ett nytt styrsystem

### 4.1 Simulator

Eftersom man hade haft undervisning i processtyrning fanns en motsvarande A&B PLC byggd som en simulator för undervisningsändamål, som jag startade upp. Studerade även de automations- och elritningar som fanns att tillgå och började strukturera upp det hela. Gick igenom den gamla programkoden för att få en bättre inblick hur den var uppbyggd och kunde sålunda utläsa vilka in- och utgångar programvaran använde sig av.

A&B PLC:s processorenhet hade tio ingångar och sex (relä)utgångar, därtill fanns två expansionsenheter med vardera tio ingångar och sex utgångar inkopplade, totalt 30 ingångar och 18 utgångar. Av dessa använde sig programvaran av 14 ingångar och 9 utgångar.

### 4.2 Relästyrningar

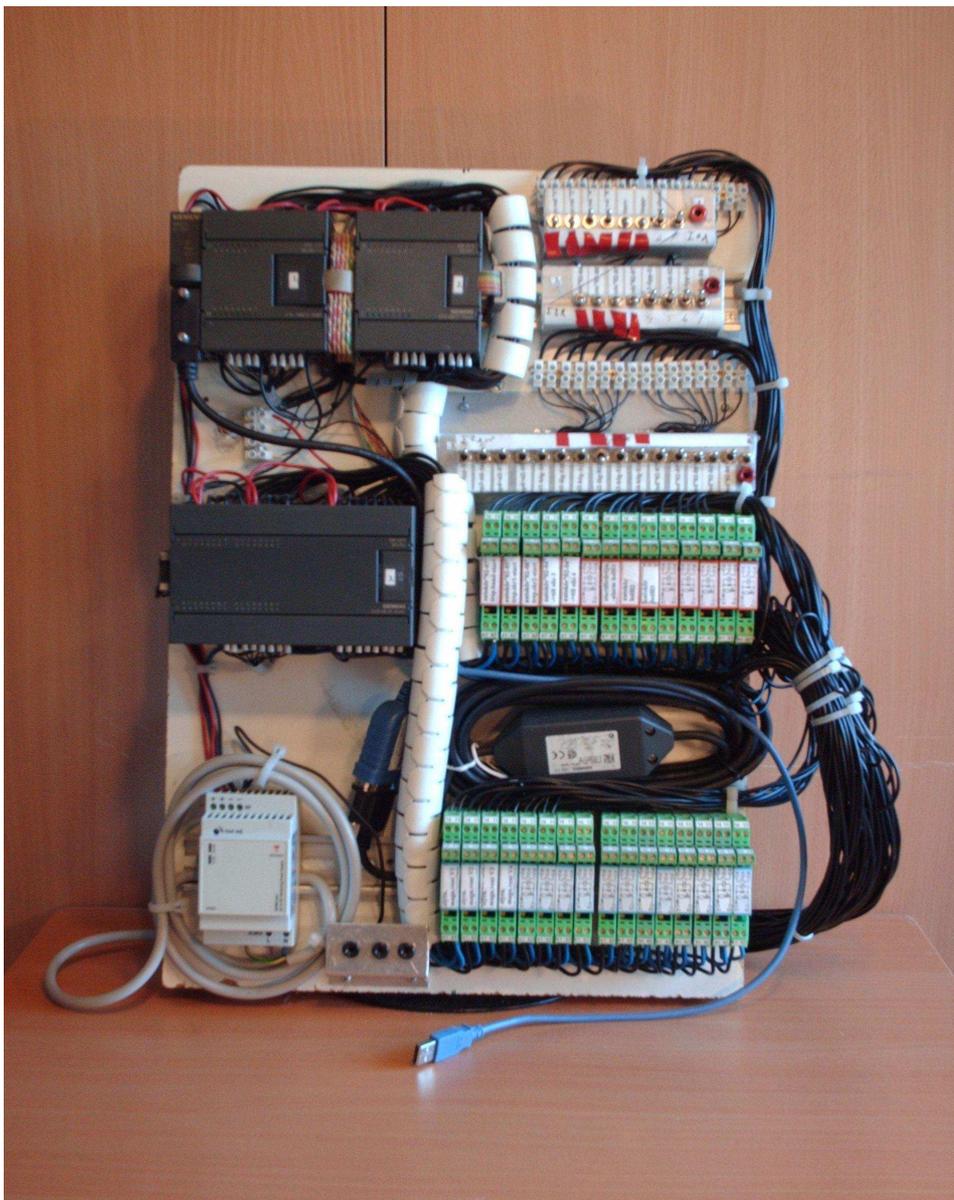
Det befintliga A&B-styrsystemet var försett med reläutgångar, dvs. man har en reläutgång som antingen är i till- eller frånläge. Den gamla A&B hade reläutgångar som kopplar in styrningarna, hur skulle man lösa det med en modern PLC med transistorutgångar?

Forskade lite i Siemens sortiment för S7-200, visst fanns det moduler med reläutgångar, men jag kom till att det blir ändamålsenligare om jag bygger upp det med separata relän.

Siemens S7-200 processorenhet har åtta ingångar och sex utgångar. Genom att införskaffa två tilläggsmoduler, en med åtta ingångar och åtta utgångar, den andra med 16 in- och 16 utgångar, så erhöles ett tillräckligt antal ingångar för att kunna ersätta den gamla styrutrustningen till fullo.

Vartefter jag funderade vidare på hur jag skulle bygga upp programkoden, blev det klart att någon typ av hjälpverktyg skulle vara till god hjälp på vägen. Byggde ihop en simulator, dels för att kunna hålla i alla trådar som behövs för programmeringen och dels för att kunna testa programkoden vartefter den framskrider.

YA hade lämpliga Phoenix relän för DIN-skena med passande 24Vdc styrning. Provkörde dessa och mätte fram att reläspolen förbrukar 22 mA per relä, så att PLC ledigt orkar mata dessa, då den ger max. 0.75A ut. Kopplade upp kontakten mellan min laptop till PLC via dess seriekabel med en USB till serieadapter. Fick kontakten att fungera och laddade upp en programsnutt för att testa om PLC:n fungerade. PLC:n kände även till tilläggsmodulerna och deras adressering.



Figur 5. Siemens S7-200 styrsystem med simulatorn

### 4.3 Styrprogrammet

I styrsystemet finns en programväljare på den separata styrpanelen, med möjlighet att välja 4 olika programkörningar. Man kan köra från valbar övre silo till valbar nedre enligt:

- 1) Körning från vänstra övre (1) till vänstra nedre silo (3)
- 2) Körning från vänstra övre (1) till högra nedre silo (4)
- 3) Körning från högra övre (2) till vänstra nedre silo (3)
- 4) Körning från högra övre (2) till högra nedre silo (4)

Det finns ingen automatik för att köra upp materialet igen från nedre silo till de övre, utan man är tvungen att göra detta manuellt med elevatorn och styra de spjäll till vilken silo man kör. Varför har man inte valt att göra en helautomatisk sekvens för hela körningen? I och med ett antal testkörningar så uppdagade jag varför så var fallet. Det spjäll som öppnar ner mot elevatorn, spjäll 3, har en alltför stor öppning. Öppnas spjället så åker hela nedre siloinnehållet ner i elevatorn och den blir igentäppt. Kanske det kunde räcka till i vårt fall, att minska på spjällöppningen eller kanske låta tillverka ett inställbart förspjäll?

Den andra bristen var att elevatorskoporna inte tömde ur tillräckligt, antingen är de för djupt utformade eller så är vänddiametern för liten uppe i elevatorn. Genom ett par kontrollfönster på elevatorn så kan man iaktta hur granulatet körs, man har en del granulat som hamnar på sidan om, det ramlar tillbaka ner igen. Detta sker oberoende av elevators hastighet, som är justerbar.

Tog kontakt med min handledare och vi tog oss en titt på problemen. Vad elevatorskoporna beträffar, konstaterade vi att elevatorn nog fungerar i stort. Eftersom motsvarande konstruktion vanligen används för att transportera något tyngre material t ex. säd i en sädestork, så torde granulatets lätthet bidra till att elevatorskoporna inte tömmer helt ur.

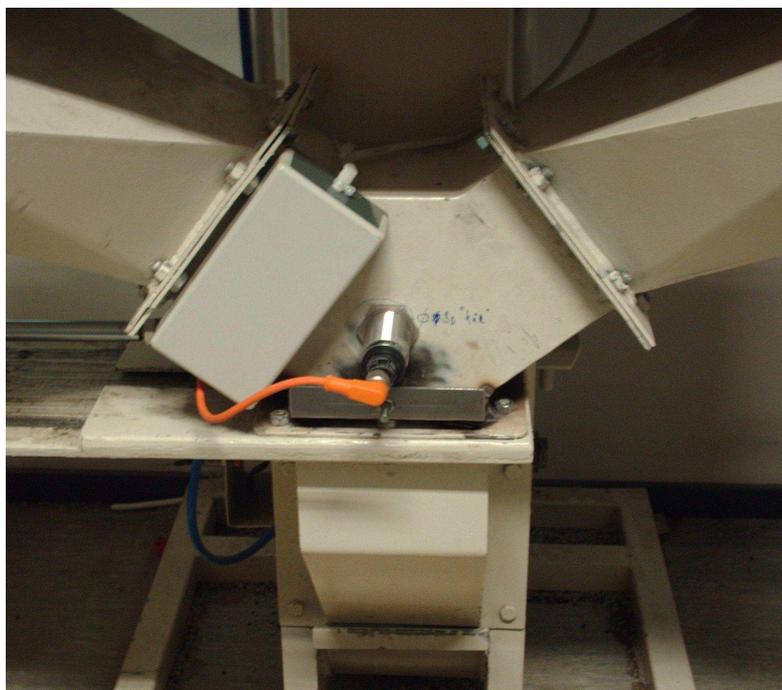
Vad beträffar det nedre spjället så kom vi till att det kunde lösas på ett antal olika sätt. Att bygga in ett nytt spjäll strax före det nedre, och så förhindra att granulatet slinker ner och gör att elevatorn går igen, verkade vara det mest användbara alternativet i detta fall. För att få information om det finns granulat nedan om spjället så borde också en givare monteras för detta ändamål.

## 5 Ombyggnad

### 5.1 Konstruktion

Monterade ner nedre siloarna och lösgjorde den konstruktion som finns ovanför det nedre spjället. Tog kontakt med YA:s metallavdelning och förde konstruktionen för ombyggnad för att få in ett justerbart spjäll. Bad dem också göra ett 30 mm hål strax ovanför det nya spjället för att kunna montera in en kapacitiv givare.

Fick konstruktionen tillbaka, färdigt ombyggd från metallavdelningen. Satte in den kapacitiva givaren och monterade ihop delarna. Började testköra och justerade in det nya spjället, så att det inte bromsade in granulatet vid en kontinuerlig körning. Spjället skulle ändå sakta upp granulatet så pass mycket att inte elevatoren blir täppt, då bägge nedre silona är fulla med granulat och det nedre spjället öppnas. Justerade in och fixerade spjället, så att det nu fungerade oberoende av körsättet. Med hjälp av ytterligare ett antal körningar så kunde den kapacitiva givaren injusteras, så att den gav en "1" då silosen fylldes med granulat, och en "0", när silon tömdes på sitt innehåll.



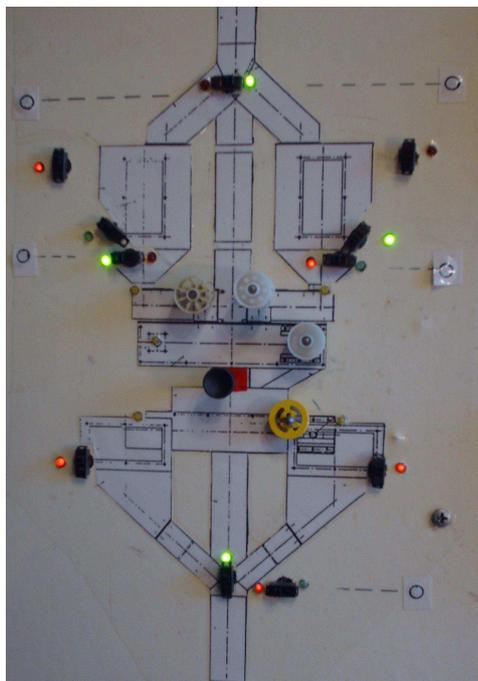
Figur 6. Spjället med givaren

## 5.2 Modellering

Då jag började fundera på hur den nya programkoden skulle börja gestalta sig, kom jag fram till att någon typ av hjälpverktyg skulle underlätta programmeringen. Byggde ihop en småskalig modell av den ursprungliga, med möjlighet att simulera modellens funktioner.

Började bygga ihop modellen genom att montera fem miniatyrelmotorer och sätta märk LED till dessa. Genom att en av motorerna skulle rotera i båda riktningarna, kopplade jag in ett relä, för att kunna byta dess körriktning. Monterade 16 minatyrbrytare för att motsvarara de givarsignaler man får från den riktiga modellen. För att få indikering i vilket läge brytarna fanns, kopplade jag en eller två LED till brytarna, beroende på den signal/nivå den ska representera. Brytarna är av ON-ON modell. Vid programmering kan brytarna ställas i de lägen som motsvarar det för tillfället aktuella programskedet.

Förbindelsen mellan modellen och simultorn blev en 32-polig kontakt vilket täckte de grundläggande in- och utsignaler man kör i sammanhanget. Trådarnas antal kan utökas vid behov. En sådan lösning ger möjlighet att koppla ifrån modell och simulator för att felsöka vid behov.



Figur 7. Småskalig modell

### 5.3 Programmering

För att programmera S7-200 logiken, så finns det en speciell seriekabel med inbyggt interface, som följer med programvaran. Programvaran är Siemens Simatic Step-7 Micro/Win. Använde en laptop med enbart usb-port, så jag blev tvungen att köra via en serie/usb-adapter, vilket fungerade alldeles utmärkt.

Jag började göra en programkod med FBD, programmering i funktionsblock. Det blev en del timers och andra logikvilkor för att få en sekvens att fungera. Gjorde så att programväljaren gav ett startvillkor samt kontrollspänningen från motorernas kontakter, det andra startvilkoret. Vid utebliven "1" från kontrollspänningen så ges alarm, som kvitteras separat.

Eftersom transportkedjan består av ett antal transportörer, så ska dessa starta en efter en, dvs. med en viss fördröjning. Detta löste jag genom att den första gav startvillkor för den följande, med en viss fördröjning. Efter att de tre motorerna som ingick i det valda programmet startats, ges signal till den styrning som öppnar det spjäll för den övre silos som ingår i resp. programkörning. Spjället stängs, antingen efter signal om att silon har tömts på sitt granulat eller signal om att den silo till vilken man kör, blivit full. Efter att spjället varit stängd i 15 sekunder, ska transportkedjan stanna. Detta sköts av en tidsfördröjning och en vippa som nollställer motorerna.

Med min ihopbyggda simulator kunde jag noll- eller ettställa de ingångar som för ögonblicket var aktuella i programmeringsskedet. Den utvecklade programvaran följer i stort de gamla programsekevenserna och i och med att det nya programmet består av FBD-block, kan man enkelt göra önskade omprogrammeringar.



## 6 Resultat

Med tanke på att modellen är avsedd att användas för undervisning, så var ett av kraven att programvaran skall vara överskådlig och entydig, för att kunna uppfylla sitt syfte. Hade avsikten varit något annat, skulle programmet ha varit uppbyggt med underprogram som ger en kortare och mera komprimerad programkod.

Programvaran innehåller 77 programblock (Network). För att få alla fördröjningstider på sina platser, så var den småskaliga modellen en verklig tillgång. Kopplade in simulatören till den riktiga processmodellen och körningen av de olika sekvenserna fungerade planenligt.

En helautomatisk prosessekvens finns inte med i den här programvaran, eftersom ett antal funktioner måste implementeras bl.a. fjärrmanöver för start av frekvensomvandlare, vilket är lite av överkurs i dagens läge, men finns som en option i I/O -listan.

## 7 Utvärdering och diskussion

Vid en första anblick kommer den nya programvarans flexibilitet inte riktigt fram, genom att man har en motsvarande körning som den gamla. Men vartefter man blir förtrogen med den moderna PLC:ns flexibilitet, så kommer programvaran och programmeringen till sin fulla rätt. Skulle gärna ha jobbat vidare med att koppla in en touch display som kunde ha möjliggjort både styrning och ett åskådliggörande av processmodellen i drift. Av ekonomiska skäl var det inte möjligt att realisera den visionen nu, kanske vid en senare tidpunkt.

Efter ett tag fick jag problem i själva programmeringen. En sekvenskörning gick bra, men då det blev en valsituation för val av program mellan 1 av 4 möjliga så låste programmet sig. Konsulterade min handledare och vi kom till att programmet körde ”fast”, då jag hade flera valsituationer. En lösning kunde ha varit ett annat programmeringsverktyg gjord för sekvensstyrningar, men eftersom vi har S7-200, så blev det att ta till ”hopp”, för att få programmet att köra, oberoende av programvalet.

I och med att processmodellen, enligt nuvarande planer, kommer att flyttas till nya utrymmen vid Kungsgården, blir en uppdatering av säkerheten kring dess användande aktuell. Ett antal skydd för de roterande axlarna och en inkapsling i ett låsbart bursprång, torde vara de minimikrav som ställs för en trygg användning i dagens läge. Även en omplacering av delarna i enheten skulle vara till fördel. Om manöverskåpet som nu finns längst bort, flyttas närmare processenheten skulle ge användaren en bättre överblick över styrfunktionerna.

Uppgiften att uppdatera styrsystemet på processmodellen har varit en mycket lärorik och arbetsdryg process. Den har gett mej goda insikter i hur komplicerat det kan vara att genomföra en uppdatering av ett styrsystem som ska skraddarsys för ett visst ändamål.

## 8 Källförteckning

Kördel, Lennart & Bengtsson, Peter. (1998)

*Styrteknik 2000*, Faktabok

Siemens S7-200 *Programmable Controller System Manual* (2002)

Allen-Bradley Handbok *Programmerbart system Bulletin 1745 SLC 100*

Pernu, Robert (1989-90)

*Konstruktion av processundervisningsapparat*

ingenjörarbete VTLA

Modellens undervisningsmaterial YA-elavd.

Block: MAIN  
 Author:  
 Created: 07/26/2010 08:39:28 pm  
 Last Modified: 08/18/2010 12:25:01 pm

Symbol	Var Type	Data Type	Comment
	TEMP		

POU Comment

**Network 1** Network Title

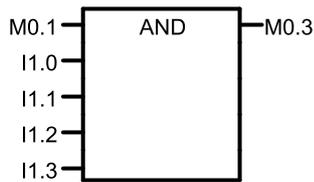
Start/Stop



Symbol	Address	Comment
Startknapp	I0.4	S2
Stoppknapp	I0.5	S3

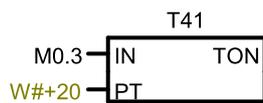
**Network 2**

Kontrollspänning från motorernas kontaktorer I1.0 - I1.3, klart för autokörning



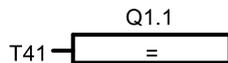
**Network 3**

Fördröjning



**Network 4**

Led H2 börjar lysa =Q1.1

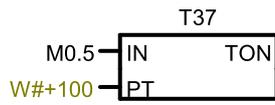


**Network 5**

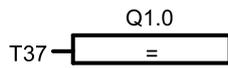
Alarm vid fel!



**Network 6**

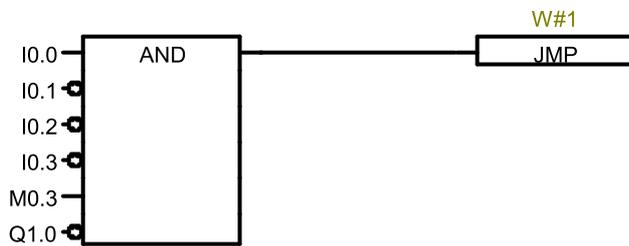


**Network 7**



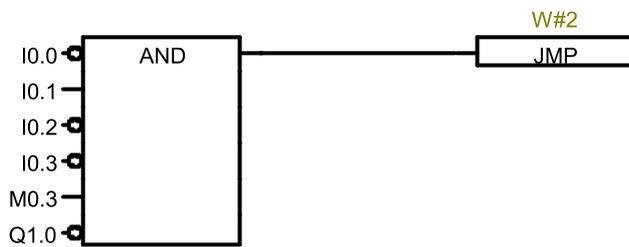
**Network 8**

programval: program nr. 1 (1000) -körning från silo 2 till silo 3



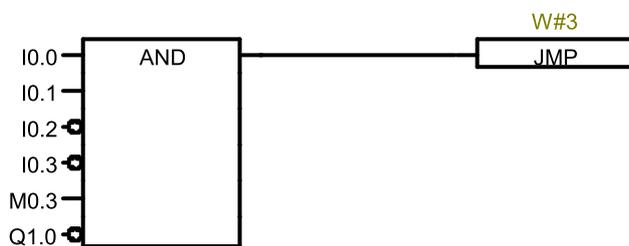
**Network 9**

programval: program nr. 2 (0100) -körning från silo 1 till silo 3



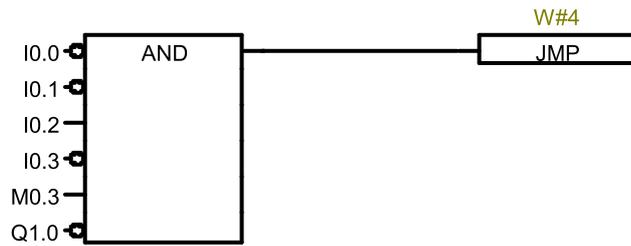
**Network 10**

programval: program nr. 3 (1100) -körning från silo 2 till silo 4



**Network 11**

programval: program nr. 4 (0010) -körning från silo 1 till silo 4

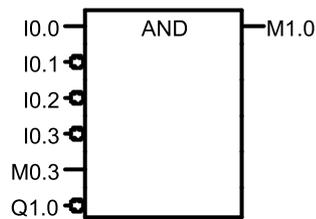


**Network 12**

program 1 start

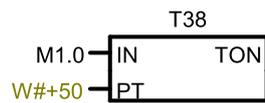


**Network 13**

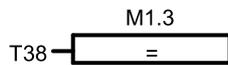


**Network 14**

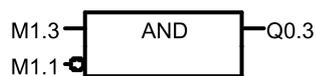
Startfördröjning motor <-02-04 = Q0.3



**Network 15**

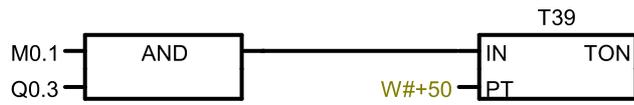


**Network 16**

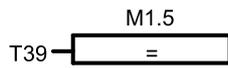


**Network 17**

Startfördröjning motor 02-07 = Q0.0



**Network 18**

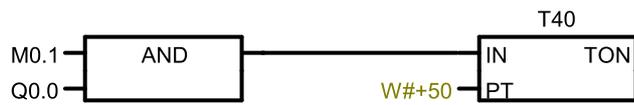


**Network 19**

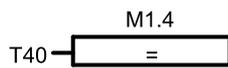


**Network 20**

Startfördröjning motor 02-08 = Q0.2



**Network 21**

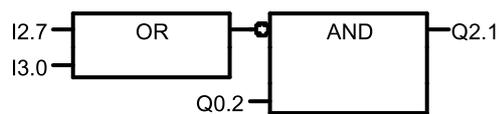


**Network 22**



**Network 23**

Spjäll 2 = Q2.1/ givare S12 och S13



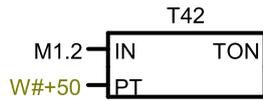
**Network 24**

När spjäll 2 stängs så stoppas motorer efter en fördröjning/ reset via stoppknapp

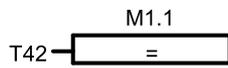


Symbol	Address	Comment
Stoppknapp	I0.5	S3

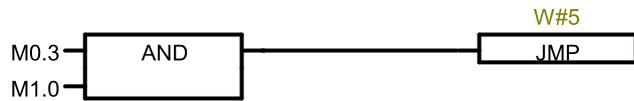
**Network 25**



**Network 26**



**Network 27**

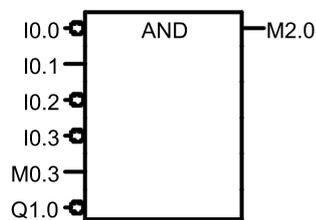


**Network 28**

program 2 start

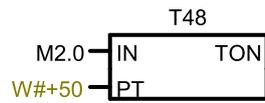


**Network 29**

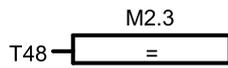


**Network 30**

Startfördröjning motor <-02-04 = Q0.3



**Network 31**

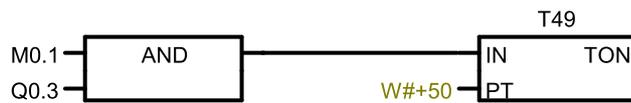


**Network 32**

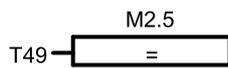


**Network 33**

Startfördröjning motor 02-07 = Q0.0



**Network 34**

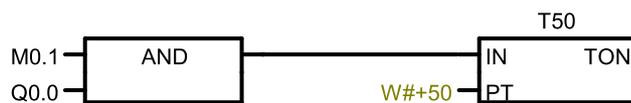


**Network 35**



**Network 36**

Startfördröjning motor 02-05 = Q0.1



**Network 37**

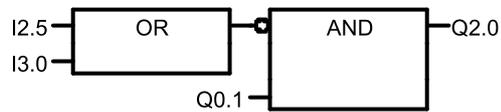


**Network 38**



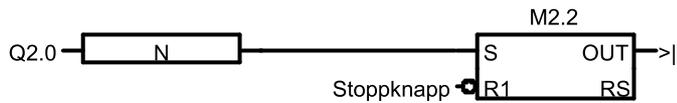
**Network 39**

Spjäll 1 = Q2.0/ givare S10 och S13



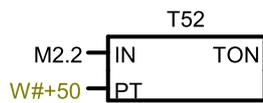
**Network 40**

När spjäll 1 stängs så stoppas motorerna efter en tidsfördröjning/ Reset via stoppknapp

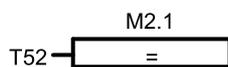


Symbol	Address	Comment
Stoppknapp	I0.5	S3

**Network 41**



**Network 42**



**Network 43**



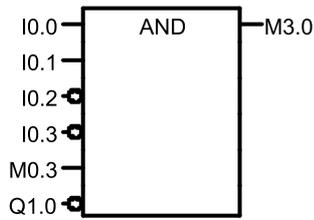
**Network 44**

program 3 start

W#3

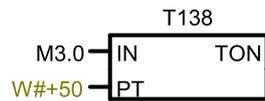


**Network 45**

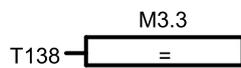


**Network 46**

Startfördröjning motor 02-04-> = Q0.4



**Network 47**

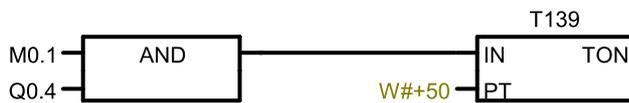


**Network 48**

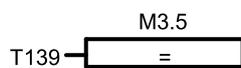


**Network 49**

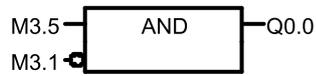
Startfördröjning motor 02-07 = Q0.0



**Network 50**

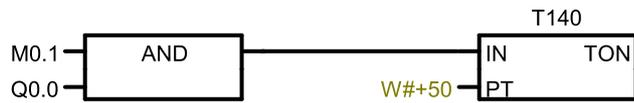


**Network 51**

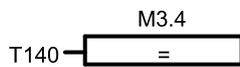


**Network 52**

Startfördröjning motor 02-08 = Q0.2



**Network 53**

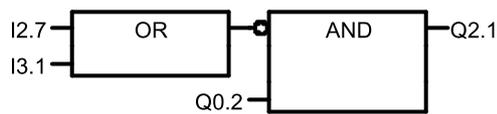


**Network 54**



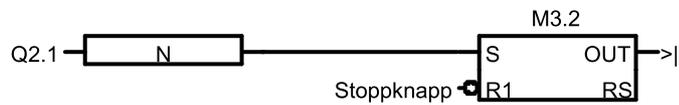
**Network 55**

Spjäll 2 = Q2.1/ givare S12 och S14



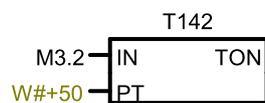
**Network 56**

När spjäll 2 stängs så stoppas motorer efter en fördröjning/ reset via stoppknapp

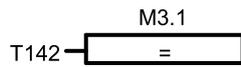


Symbol	Address	Comment
Stoppknapp	I0.5	S3

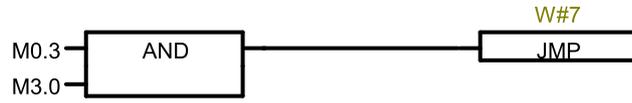
**Network 57**



**Network 58**



**Network 59**

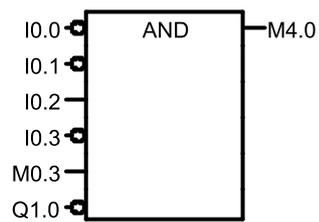


**Network 60**

program 4 start

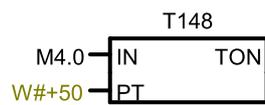


**Network 61**

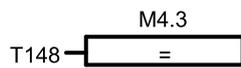


**Network 62**

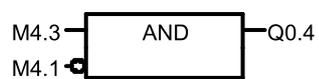
Startfördröjning motor 02-04-> = Q0.4



**Network 63**

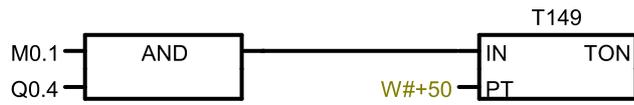


**Network 64**

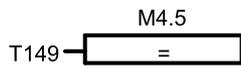


**Network 65**

Startfördröjning motor 02-07 = Q0.0



**Network 66**

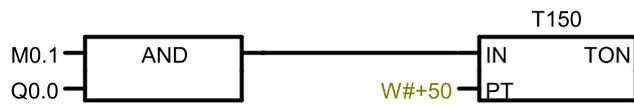


**Network 67**

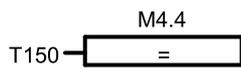


**Network 68**

Startfördröjning motor 02-05 = Q0.1



**Network 69**

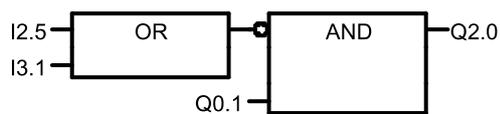


**Network 70**



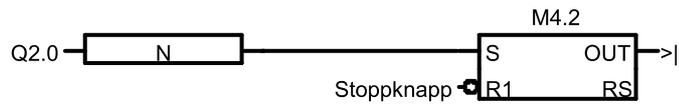
**Network 71**

Spjäll 1 = Q2.0/ givare S10 och S14



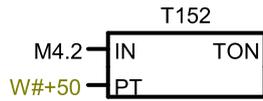
**Network 72**

När spjäll 1 stängs så stoppas motorer efter en fördröjning/ reset via stoppknapp

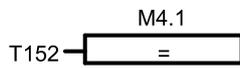


Symbol	Address	Comment
Stoppknapp	I0.5	S3

**Network 73**



**Network 74**



**Network 75**



**Network 76**



**Network 77**



Block: SBR\_0  
Author:  
Created: 07/26/2010 08:39:28 pm  
Last Modified: 07/31/2010 05:10:24 pm

Symbol	Var Type	Data Type	Comment
EN	IN	BOOL	
	IN		
	IN_OUT		
	OUT		
	TEMP		

POU Comment

**Network 1** Network Title

Network Comment

Block: INT\_0  
Author:  
Created: 07/26/2010 08:39:28 pm  
Last Modified: 07/31/2010 05:10:24 pm

Symbol	Var Type	Data Type	Comment
	TEMP		

POU Comment

**Network 1** Network Title

Network Comment