

DATORSIMULERING AV DISKRETA FÖRLOPP I TORPEDTUBER

GÖRAN PETERSSON

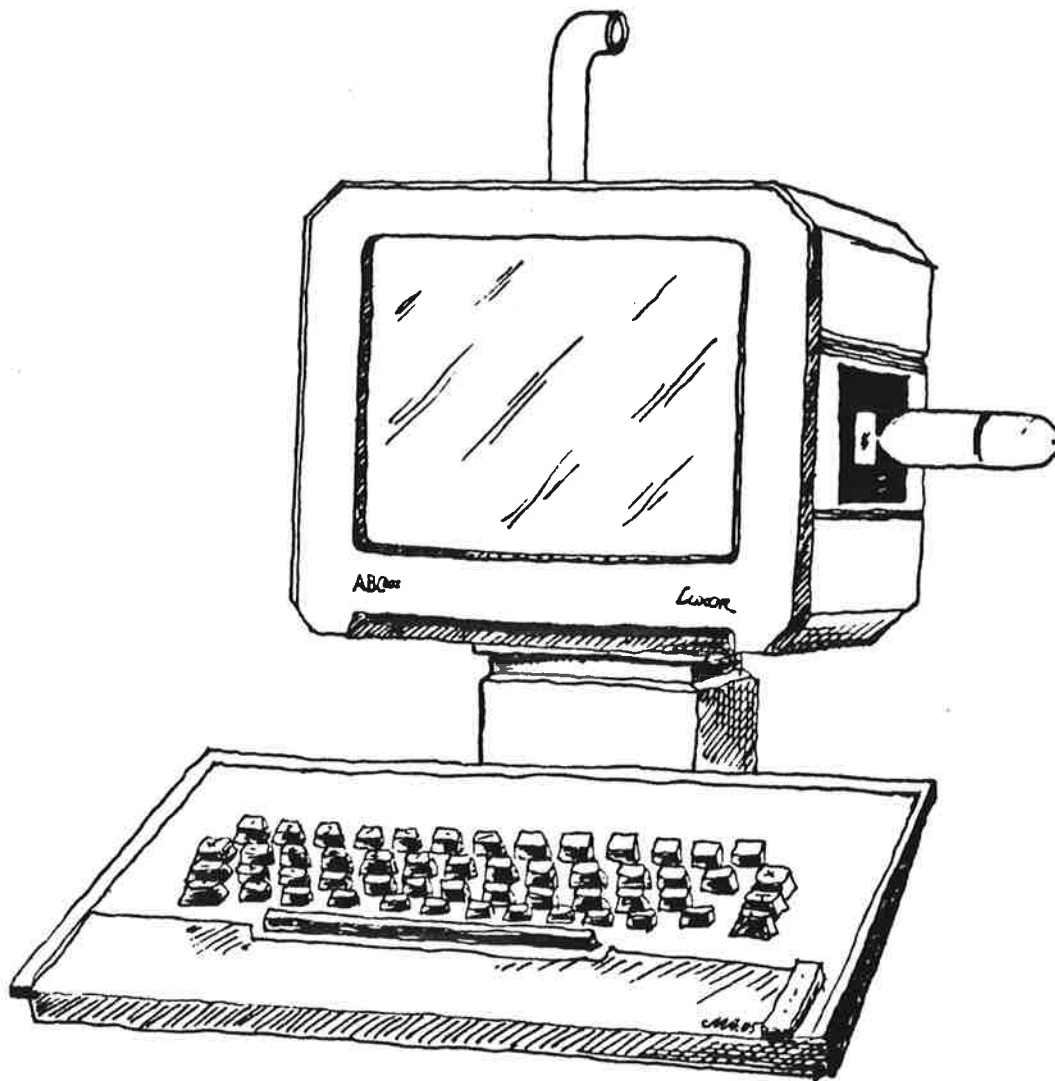
INSTITUTIONEN FÖR REGLERTEKNIK
LUNDS TEKNISKA HÖGSKOLA
AUGUSTI 1985

LUND INSTITUTE OF TECHNOLOGY DEPARTMENT OF AUTOMATIC CONTROL Box 118 S 221 00 Lund Sweden		Document name Master thesis	
		Date of issue August 1985	
		Document number CODEN=LUTFD2/(TFRT-5330)/1-056/(1985)	
Author(s) Petersson Göran		Supervisor Anders Ljungman, Gustaf Olsson (Regl.)	
		Sponsoring organization	
Title and subtitle Datorsimulering av diskreta förlopp i torpedtuber. Simulation of discrete events in torpedo tubes.			
Abstract <p>Examensarbetet omfattar framtagning av ett program som simulerar tre torpedtuber ombord på ubåtar typ Västergötland. Simuleringsprogrammet är skrivet i Basic för en ABC-802 dator. Programmet skall användas som testprogram för tubstyrssystemet. Arbetet är utfört vid Kockums AB i Malmö.</p> <p>This thesis describes the development work of a program for the simulation of three torpedo tubes of the Västergötland submarines. The simulation is written in Basic for a ABC-802 computer. The program will be used as a test aid for the tube software. This work was performed at Kockums AB, Malmö.</p>			
Key words Simulering Simulation			
Classification system and/or index terms (if any)			
Supplementary bibliographical information			
ISSN and key title			ISBN
Language Swedish	Number of pages 56	Recipient's notes	
Security classification			

DOKUMENTTABLAD RT 3/81

Distribution: The report may be ordered from the Department of Automatic Control or borrowed through the University Library 2, Box 1010, S-221 03 Lund, Sweden, Telex: 33248 lubbis lund.

DATORSIMULERING AV DISKRETA FÖRLOPP I TORPEDTUBER



EXAMENSARBETE UTFÖRT AV
GÖRAN PETERSSON

Mottagare

FMV 3 ex
KBS ALJ

Lunds Tekn Högskola 4 ex

Göran Petersson 4 ex

För kännedom

Spr LS KH

UB TYP VGD. DATORSIMULERING AV DISKRETA FÖRLOPP I TORPEDTUBER

SAMMANFATTNING

Examensarbetet omfattar framtagning av ett program som simulerar tre av ub typ Västergötlands torpedtuber. Simuleringsprogrammet är skrivet för en dator av typ ABC-802.



Malmö 85-09-02


FMV

Malmö 85-09-02


KAB

Mottagare

INNEHÅLL

	Sid
1. INLEDNING	3
1.1 Bakgrund/syfte	3
1.2 Problembeskrivning	4
1.3 Utförande	5
1.4 Rapportens uppläggning	5
2. PROGRAMBESKRIVNING	6
2.1 Slutversion	6
2.2 Utveckling	7
2.3 Exekvering	8
2.4 Ventilhantering	8
2.5 Tidsfördröjningar	9
2.6 Testfunktion	11
3. FRAMTIDA UTVECKLINGSMÖJLIGHETER	12
APPENDIX:	
A. PROGRAMVARUDOKUMENTATION INKL FLÖDESSCHEMAN	
A.1 Variabeldeklarationer	
A.2 Uppstart	
A.3 Initiering av huvudprogram	
A.4 Huvudloop	
A.4.1 Kontroll av säkringar	
A.4.2 Utmaskning av ventiländringar	
A.4.3 Kontroll av tidssubrutiner	
A.4.4 Tangentbordsavläsning	
B. FYSISKT SYSTEM	
B.1 In/ut-kort	
B.2 Givarmatriser	
B.3 Bildskärm	
C. PROGRAMKÖRNING	
C.1 Operatörsinstruktioner	

Mottagare

1. INLEDNING

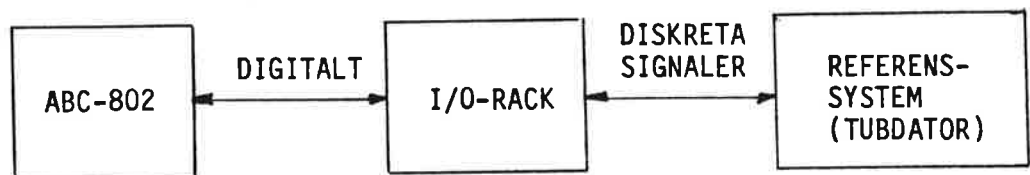
Examensarbetet omfattar framtagning av ett program som simulerar tre av ub typ Västergötlands torpedtuber. Simuleringsprogrammet är skrivet för en dator av typ ABC-802.

1.1 Bakgrund/syfte

Simuleringsprogrammet är skrivet som ett examensarbete vid Kockums AB i Malmö. Institutionen för Reglerteknik vid Lunds Tekniska Högskola har svarat för skolans insatser. FMV har bidragit med råd och finansiering.

Programmet skall användas som hjälpmedel då programmen till tubdatorn, ingående i tubstyrssystemet, skall skrivas.

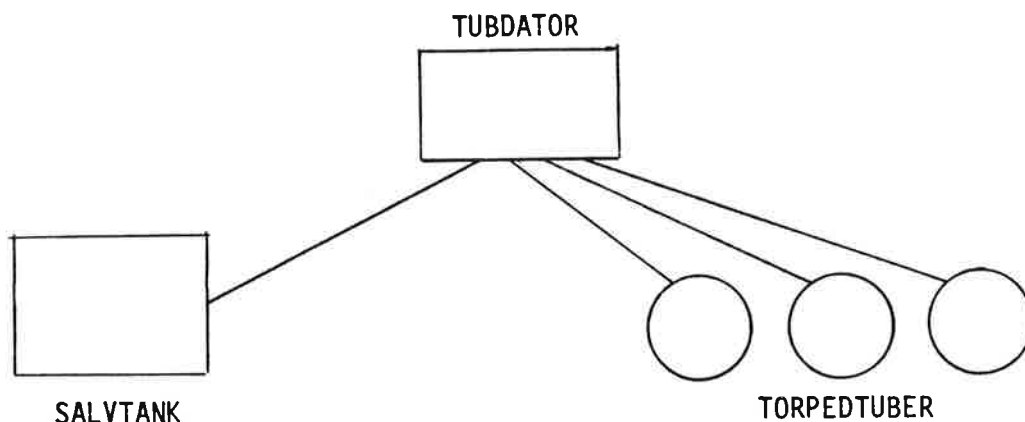
Tubdatorn skall kunna styra ett antal torpedtuber och tankar. Programmen till tubdatorn skall utvecklas på ett referenssystem.



ABC-802 bildar tillsammans med kortracken ett system som ur tubdatorns synpunkt i huvudsak är identiskt med det verkliga fysiska systemet.

1.2 Problembeskrivning

Tubdatorn ombord på ub typ Västergötland (ub typ Vgd) skall styra och övervaka ett antal torpedtuber och tankar. Tubdatorn kommer att arbeta i reell tid, vilket medför att simulatormåste arbeta inom samma tidsgränser som det fysiska systemet. I simuleringen ingår tre tuber och en saltvattentank enligt nedanstående figur.



De tre torpedtuber som skall simuleras styrs i huvudsak av ett antal elektrohydrauliska ventiler. Dessa ventiler är av on/off-typ och kan därmed endast anta två lägen. Vissa moment utförs helt manuellt. För indikering av lägen på ventiler, luckor och lås m m samt för indikering av nivå i tuber och saltvattentank finns ett antal binära signaler. Principskisser över tuber och saltvattentank återfinns i Appendix B.

Ventilstyrningen skall kunna utföras antingen från ABC-802s tangentbord eller från referenssystemet. På ABC-802s bildskärm skall man kunna se ventilstatus, givarstatus, tubtryck, felutskriften samt tillåtna kommandon. Fyllning/tömning av tuber samt öppning/stängning av luckor skall kunna följas på bildskärmen.

Fel i tubdatorns program, som ej upptäcks under utvecklingsstadiet, blir dyra att åtgärda senare och kan medföra fara ombord på ubåten. Följaktligen måste simulatormåste efterlikna det fysiska systemet så bra som möjligt. Simulatormåste även utföra en mängd tester och kontroller av givna order.

Mottagare

1.3 Utförande

Simuleringen utförs med en ABC-802 som via en korttrack kopplas till referenssystemet.

Fördelar: Billig simulator, relativt snabb, tillräckligt bärbar för att kunna tas ned i en ubåt (vilket kan komma att behövas vid felsökning/testning), finns ett färdigutvecklat I/O-kortsystem.

Nackdelar: Litet minne (i praktiken 27 kbyte) som ej går att bygga ut med bibehållen snabbhet, datorn är ej realtidsprogrammerbar, programmet får skrivas i BASIC.

För att få ett snabbt program har allt "arbete" lagts i subrutiner. Huvudloopen sköter i stort sett endast den administrativa delen. I avsaknad av en realtidsskärna måste i huvudloopen ingå de vanliga operativsystemfunktionerna "tidsstyrd väntan" (WAITTIME) samt "vänta på händelser" (WAITEVENT).

Eftersom minnesutrymmet är begränsat är programmet uppdelat i två delar. Den första delen ritar upp skärmen, sätter vissa flaggor enligt operatörens önskemål samt ger information om programmet. Uppstartningsprogrammet avslutas med ett anrop av huvudprogrammet som då hämtas in till primärminnet och börjar att exekveras. Den första programdelen försvinner då ur minnet.

1.4 Rapportens uppläggning

Övergripande information finns i kapitel 1. De mer detaljerade programbeskrivningarna finns i kapitel 2. I kapitel 3 finns några idéer om hur man kan vidareutveckla programmet.

Appendix A innehåller flödesscheman enligt Nassi-Schneiderman. Listor på in/ut-signaler kan man hitta i Appendix C.

Mottagare

2. PROGRAMBESKRIVNING

2.1 Slutversion

Hela simuleringsprogrammet ryms ej i datorn vilket medför att programmet är uppdelat i två delar.

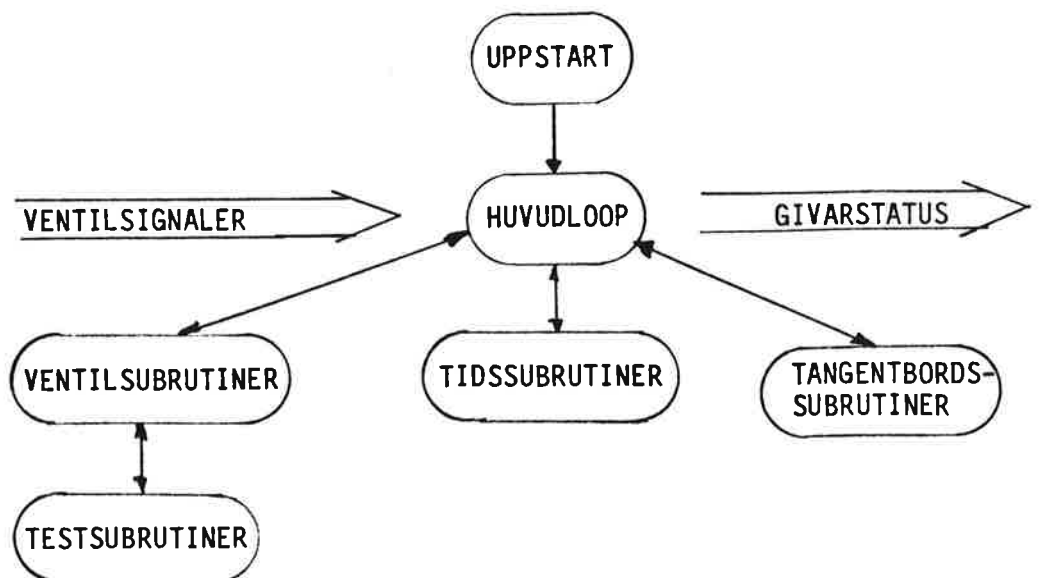
I programmet "UPPSTART" ges operatören möjlighet att:

- välja vilka tuber som skall simuleras
- välja aktuellt ubåtsdjup
- välja ventilernas styrsätt
- välja vilka skvalptester som skall utföras
- få upp programinformation på skärmen.

Därefter ritas den grafiska bilden upp på skärmen och huvudprogrammet anropas.

Programmet "HUVUD" består av en kort huvudloop och 78 st subrutiner. Huvudloopen försöker efterlikna en realtidskärna. Den avkänner inportar, skickar status via utportar, avläser tangentbordet och övervakar en kö med tidsfördröjningar. Om loopen exekveras tillräckligt snabbt blir det god överensstämmelse med det verkliga systemet som arbetar i reell tid.

I huvudloopen jämförs ventilsignaler från inportarna med intern ventilstatus. Eventuella skillnader maskas ut och medför ett anrop av aktuell subrutin. En kontroll sker om någon av de aktiverade tidssubrutinerna har väntat sitt tidsintervall och därmed skall anropas. Tangentbordet avläses för att se om någon av de på skärmen angivna tangenterna har tryckts ned. I så fall sker ett anrop av angiven subrutin. Även kontroll av säkringar och utmatning av givarstatus sker i huvudloopen. Vid inläsning av ventilsignaler och utmatning av givarstatus sker en fördröjning ett visst antal av huvudloopens varv för att simulera fördröjningar i det verkliga systemet.



Subrutinerna är uppdelade i fyra huvudtyper.

I Ventilsubrutinerna ändras de givare som indikerar öppen/stängd ventil samt i vissa fall sätts flaggor och ges starttider till tidssubrutiner. Det sker även en kontroll av de villkor som skall vara uppfyllda för att ventilen skall få öppnas.

Tidssubrutinerna används till händelser som tar tid, t ex öppning av luckor och fyllning av torpedtuber. När en viss flagga är satt ser huvudloopen till att subrutinen anropas med vissa fastställda tidsintervall.

Tangentbordssubrutinerna anropas då någon av de på bildskärmen angivna tangenterna har tryckts ned. I de flesta fall skrivs det därefter en ledtext på nedersta raden. Operatören kan där se hur han kan precisera sitt kommando. Den subrutin som väntar på att kommandot skall fullföljas markeras med inverterad text. Programmet kan endast avläsa det senaste tangentnedslaget, vilket medför att kommandona måste ges i flera steg. Ett antal flaggor ger programmet möjlighet att fortsätta exekveringen samtidigt som kommandona ges.

Testsubrutinerna används för att simulera felaktiga signaler från tubernas nivågivare. Dessa signaler kan uppstå p g a stänk både vid fyllning och tömning av torpedtuberna. Dessa subrutiner anropas av ventilsubrutinerna.

På skivan medföljer en tidig programversion med namnen "FRAM" och "BAK". Dessa båda program innehåller kommentarer och är skrivna med avsikten att andra skall kunna förstå programmet. Den slutgiltiga programversionen är p g a begränsat minnesutrymme en komprimerad textmassa utan kommentarer. Dock är radnumreringen i de olika versionerna i stort sett densamma. Därmed kan man dra nytta av kommentarerna i den gamla versionen när man vill fördjupa sig i slutversionen. Det gäller dock att tänka på att "FRAM" och "BAK" inte är provkörda eller uppdaterade och därmed innehåller många småfel.

2.2 Utveckling

Ett antal funktionsbeskrivningar har använts för att få en uppfattning om hur tuberna fungerar. Där beskrivs med ord bl a vilka kontroller som skall utföras samt i vilken ordning som ventilerna skall öppnas. Dessa funktionsbeskrivningar skrevs om till en mer "matematisk" form och utökades med givarnas uppförande. Nästa steg blev att göra tabeller som fungerade åt motsatt håll, dvs vad som händer när en ventil öppnas, vad som måste vara uppfyllt samt när den skall öppnas. Efter diverse tester på ABC-802 skrevs programmet ned i form av Nassi-Schneiderman-diagram.

Programmet skrevs ursprungligen för att bli så snabbt som möjligt. Första programversionerna skrevs med kommentarer och tydliga rubriker. Denna goda programmeringsmetodik följdes ända till dess programbitarna till slut länkades samman. Primärminnet visade sig då rymma endast hälften av det ambitiösa

Mottagare

programmet. På ABC-802 går det inte heller att utvidga primärminnet med bibehållen snabbhet. Enda lösningen var att krympa ner programmet. Allt som kunde lyftas ut flyttades till ett uppstartningsprogram som fick namnet "UPPSTART". Det återstående programmet "HUVUD" förlorade sina kommentarer, fick sina rubriker krympta till ett minimum samt utsattes för en mängd av förenklingar. Det slutliga programmet är därför mycket svårläst men utför samma uppgifter som det ursprungliga.

2.3 Exekvering

Då datorn får kommandot "RUN UPPSTART" läses första delen av programmet in från flexskivan. När operatören har svarat på vissa frågor ritas den grafiska bilden upp på skärmen. Därefter läses andra delen av programmet in och den första delen försvinner ur datorn.

Efter vissa initieringar går programmet in i en huvudloop som därefter genomlöps kontinuerligt. I huvudloopen kontrolleras ventilernas lägen varje varv och om någon ventil har ändrat sig sedan föregående varv sker ett hopp till en bestämd subrutin. Huvudloopen administrerar även de tidsfördröjnignar som behövs vid bl a öppning av luckor. Det enda som bryter loopen (förutom subrutinhopp och Ctrl C) är kommandon som medför att ubåten vattenfylls. Datorn nonchalerar då det sist givna kommandot och hoppar till en yttre slinga där den väntar på signal från operatören att fortsätta exekveringen.

När programmet exekveras är det operatören alternativt operatören och en extern dator som avgör vad som skall hända. På skärmen kan man se ventilstatus och givarstatus samt följa vad som sker med tre grafiska tuber.

Varningar och felpåpekanden skrivs ut på den nedersta raden.

2.4 Ventilhanteringen

Tubdatorn styr ventilerna via ett antal ledningar vars spänningsnivå indikerar TILL/FRÅN. Dessa spänningsnivåer kommer in i kortracken där två "DATABOARD-kort" omvandlar spänningarna till 8-bitars tal med TTL-nivåer. Dessa 8-bitars tal kan avläsas via en för ABC-802 och kortracken gemensam databuss.

Ventilsignalerna hanteras i programmet i form av bytes. Varje ventil motsvaras av en bit. Etta (1) står för öppen ventil och en nolla (0) representerar en stängd ventil. I Appendix B framgår hur ventilerna är fördelade i respektive byte.

De 8-bitars tal som hämtas från kortracken läggs i vektorn "HÄMTV". Om operatören i uppstartningsprogrammet väljer tangentbordsstyrning kan man via tangentbordet ändra direkt i "HÄMTV". Det går endast att styra på ett sätt. Antingen utifrån eller via tangentbordet. "HÄMTV" kopieras över till "WAITIN" för att simulera fördröjningar i det verkliga systemet. Den rad i matrisen "WAITIN" som väntat ett visst antal varv kopieras över till vektorn "NYV".

De ventilsignaler som ligger i "NYV" jämförs därefter med programmets interna ventilstatus, vilken finns i "GV". Eventuella skillnader maskas fram och därmed vet programmet vilka subrutiner som skall exekveras. Därefter uppdateras den interna ventilstatusen genom att "NYV" kopieras över till "GV".

2.5 Tidsfördröjningar

I det verkliga tubsystemet förekommer det ett antal tidsberoende händelser. Till exempel när tuberna vattenfylls tar det en viss tid från det att ventilerna har öppnats till dess att vattennivån når nivågivarna. Det går naturligtvis inte att låta processorn stå och vänta i ett program som simulerar ett system i reell tid.

Med hjälp av ett antal flaggor håller programmet ordning på totalt 22 olika tidsberoende händelser. Flaggorna motsvarar det "WAITTIME" som är vanligt i datorer som klarar realtidsprogrammering. ABC-802 kan inte realtidsprogrammeras. Det enda hjälpmedel som finns är en intern klocka.

I huvudloopen sker en kontroll av flaggvektorn "IGÅNG". Om något element är "SANT" jämförs "STARTTID(T)" med den interna klockan. Om skillnaden mellan dessa tider överstiger "INTERVALL(T)" sker ett anrop av aktuell tidssubrutin. I subrutinen ges en ny starttid eller också nollställs "IGÅNG(T)" om t ex tuben är fylld. Ibland ändras även "INTERVALL(T)" inuti subrutinen.

Vid tidsjämförelserna används endast hela sekunder. "INTERVALL(T)" begränsas därför till intervallet 1-59 s. Storleken på intervallen styrs även av grafiska begränsningar samt av krav på precision då händelsen avbryts och påbörjas i motsatt riktning.

Vattennivån i tuben visas grafiskt i fem diskreta steg. Programmet arbetar emellertid med högre upplösning motsvarande 16 steg.

Här nedan följer en kortfattad förklaring till några av de tidsintervall som används i programmet.

Fyllning av Salvtank

Intag av vatten till salvtanken sker då ventil V18 öppnas. Mängden vatten i tanken uppdateras med ett tidsintervall (TID 17) som beräknas ur angivet djup. Flödet i ventilen varierar nämligen med djupet.

Fyllning och länsning av tub

Vattenmängden, som ryms i en tom tub, motsvarar 16 diskreta nivåer. Beroende på om torped finns i tuben eller ej använder programmet olika tidsintervall för uppdatering av nivån i tuben under fyllning och länsning. Vid en uppdatering ökas/minskas nivån med ett diskret steg. För att även kunna ta hänsyn till det vatten som finns i luftavlopps- och länsledningarna tillkommer två diskreta nivåer för länsledningen och fyra för luftavloppsledningen. Därför kan variabeln NIVÅT (TUBNR) anta värdena -2 till 20. Tidsintervallet som motsvarar fyllning/länsning av ett diskret steg, kallas INTERVALL(T). T är ett index som anger vilken tidssubrutin som används.

Öppning och stängning av luckor

Luckornas lägen indikeras grafiskt med 5 olika positioner. Internt i programmet sker öppning och stängning i tolv steg. Luckornas lägen uppdateras med ett bestämt tidsintervall på samma sätt som vid fyllning och länsning.

2.6 Testsubrutiner

Vid tömning och fyllning används tre nivågivare för att indikera fylld resp tömd tub. Två av givarna sitter i luftavloppet ovanför tuben och den tredje sitter i länsledningen under tuben. Det finns risk för att vatten skvalpar upp på givarna och indikerar en felaktig nivå. Programmet i tubdatorn kommer att ta hänsyn till detta och kräva indikation under vissa tidsintervall för att den skall acceptera nivån. Tubdatorn tillåter inte heller hur många "stänkindikationer" som helst.

Mina testsubrutiner i ABC-datorn avser att "testa" att kontrollerna utförs enligt givna specifikationer. I uppstartningsprogrammet kan operatören ange vilka testsubrutiner som skall aktiveras, t ex om test 1 aktiveras kommer tubdatorn att få två falska indikeringar när tub 1 fylls.

I programmets grundversion ligger en variant av de många tester som kan göras. Testsubrutinerna är skrivna så att det är enkelt att variera testet. Förutom att variera antalet "stänk" kan man ändra givarnas nivåer och därigenom få dem att indikera med olika tidsintervall samt i valfri ordning. Observera att ändringar av testsubrutinerna endast kan ske genom ändring i programmet. Aktivering av testerna sker dock i början av varje programkörning.

De olika testsubrutinerna används i följande fall:

Test 1	Fyllning av tub 1
Test 2	Fyllning av tub 2
Test 3	Fyllning av tub 3
Test 4	Tömning av tub 1
Test 5	Tömning av tub 2
Test 6	Tömning av tub 3

3. FRAMTIDA UTVECKLINGSMÖJLIGHETER

Programmets storlek försvårar framtida modifieringar. Slutversionen av programmet "HUVUD" fyller helt upp primärminnet. Något måste alltså strykas för att en ny funktion skall få plats. Salvtankarna är en relativt fristående del av programmet som skulle kunna plockas bort - det ger dock ingen större minnsvinst. Det finns också en liten möjlighet att "LUXOR" utvecklar en kraftfullare dator som är kompatibel med ABC-802.

Bortsett från de praktiska problemen kan man tänka sig att använda simulatören till utbildning av ubåtspersonal. Programmet är i nuvarande form användbart vid utbildning. Dock krävs det att man lägger ner lite tid på att förstå hur programmet fungerar. Om man förenklar ventilstyrningen från tangentbordet, lägger in fler ledtexter samt skriver en förenklad version av operatörsinstruktionerna så kommer programmet att vara lämpligt för utbildning.

Mottagare

A. PROGRAMVARUDOKUMENTATION INKL FLÖDESSCHEMAN

A.1 VARIABELDEKLARATIONER

In/ut-hantering, sker med bytes

WAITIN (0:4,8)	Fördröjningsmatris som ger 5 varvs fördröjning för insignaler.
WAITOUT (0:4,8)	Fördröjningsmatris för utsignaler.
VRAD	Den rad i WAITIN som läses av Nyv. Hämtv skriver på (VRAD-1).
GRAD	Den rad i WAITOUT som läses av kortracken. Givarändringar läggs i (GRAD-1).
NYV(8)	Hämtar ventilinsignaler från WAITIN (NYVEKTOR).
GV(8)	Innehåller gällande ventilstatus (GAMMALVEKTOR).
HÄMTV(8)	Hämtar ventilinsignaler från kortracken. Läger dem i WAITIN.
BYTE	Loopindex vid utmaskning av ventiländringar.
DIFF	Används vid utmaskning av ventiländringar.

Tubstatus

NIVÅT(3)	Nivå i tub (-2...20) varav (0..16) ligger inuti tuben.
TRYCK(3)	Tryck i tub. Finns UBÅTSTRYCK HAVSTRYCK samt TRYCKLUFTSTRYCK (0, 8, 16).
LADDAD(4)	Anger torpedtyp i tub. Nolla anger tom tub.
AKTIVERAD(2)	Anger om torpeden är aktiverad. Gäller endast vissa torpedtyper.
LFORML(3)	Läge formlucka. 5 grafiska lägen, 12 interna.
LMYNNL(3)	Läge mynningslucka. 5 grafiska lägen, 12 interna.

Mottagare

Tidsföröjning

IGÅNG(22)	Logisk variabel som anger om tidssubrutin nr (index) är igång
STARTTID(22)	Anger när tidssubrutin nr (index) började vänta. Endast hela sekunder
INTERVALL(22)	Den tid som skall gå mellan varje exekvering av tidssubrutin nr (index)
VARV(10)	Varvräknare till tidssubrutinerna FYR samt subrutinerna TEST
FYRTID(4,4)	Vid avfyring går givare 65 "1-0-1-0-1". De fyra tidsintervallen mellan omslagen ligger i matrisen Fyrtid
TID	Hjälpvariabel vid kontroll av om någon tidssubrutin väntat färdigt
T	Index i tidsfördröjningsvektorer

Variabler som är gemensamma för programmen START och HUVUD (Common-deklarerade)

SG(98:119)	Givare som förekommer på endast ett ställe. Givarnas funktion framgår av princip-scheman, se Appendix E.
TEST(1:6)	Logiska variabler som anger om skvalptest skall utföras. TEST(1..3) Utförs vid fyllning av respektive tub TEST(4..6) Utförs vid tömning av respektive tub
TID17	Tidsvariabel som anger hur snabbt saltanken skall fyllas. Tiden beror på aktuellt djup.
RACK	Om RACK = -1 sker styrningen av ventilerna utifrån, via kortracken. Annars ges ventilsignalerna ifrån tangentbordet.

Övrigt

S(3,51:97)	Givarmatris. S(tubnr, givarnr) Ex S(1,57) Givare 57, Tub nr 1
	I dokumentationen utelämnas oftast TUBNR. De flesta subrutiner används av alla tre tuberna Ex S(57) = S(TUBNR,57)

Mottagare

STOPP(6)

Används för att bryta skvalptesterna. Därmed kan vi lätt välja hur många gånger nivågivarna skall ge tillfälliga nivå-indikationer.

POS(3)

Anger i vilken grafisk punkt tublocket till respektive tub är. Anger läge i X-led.

V21, V22, V23

Etta anger att ventilen är öppen. Används för att öka programmets läsbarhet. Ventilererna är handmanövrerade.

NIVÅS

Nivå i salttank. Använder internt 10 olika nivåer. De fem nivågivarna SG(101..105) visar nivån i salttanken.

X,Y,Z

Slaskvariabler. Används främst till programförenklingar. Förekommer endast i direkt anslutning till sina tilldelnings-satser.

I,J,K

Loopindex

F1, L1, Ä1, K1, S1,
S2, S3, X1, X2, X3,
T1, T2, T3, Y1, Y2,
Z1, Z2

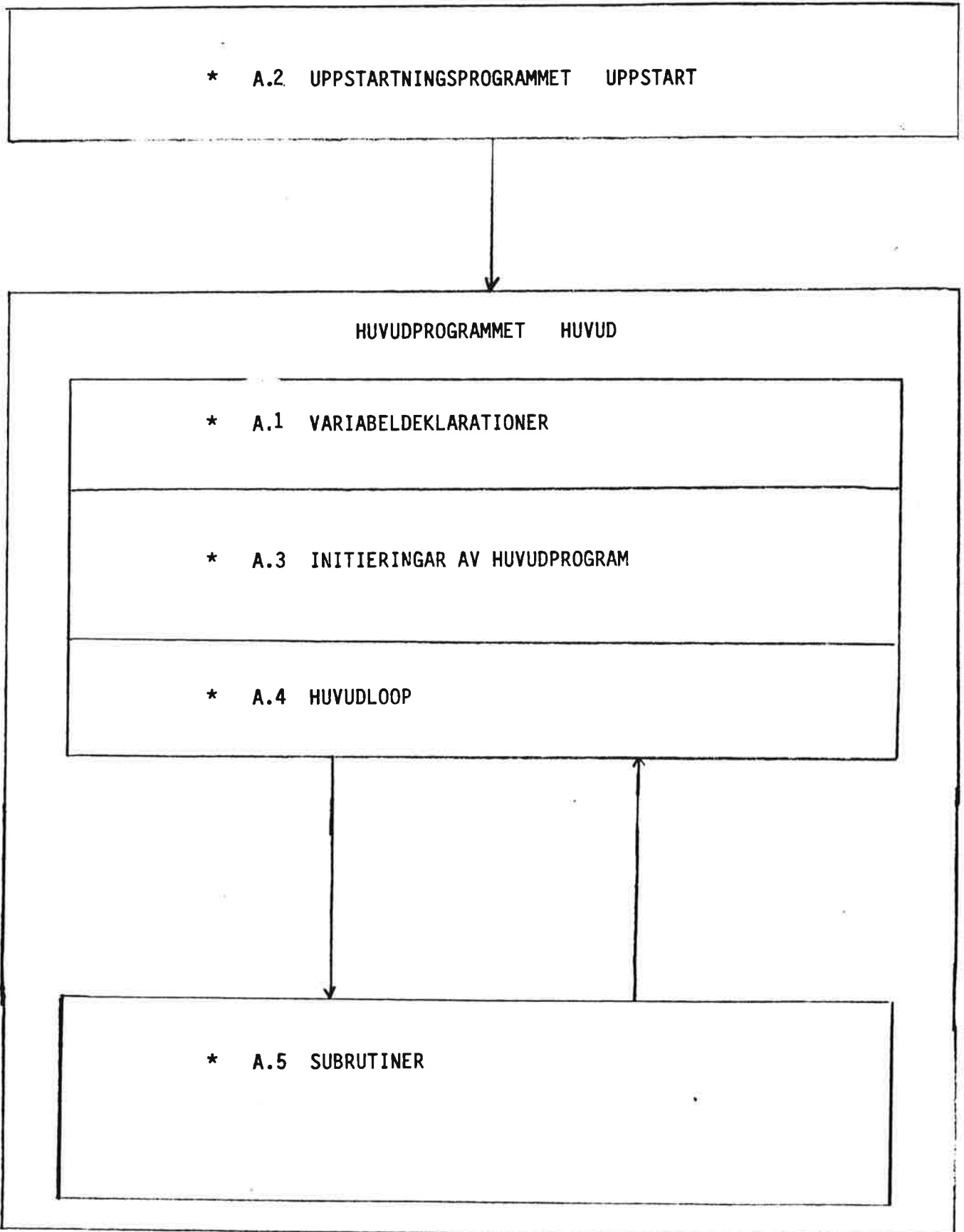
Denna kombination (en bokstav, en siffra) används som "flaggor" vid ingrepp via tangentbordet

NIVÅ 161, NIVÅ 162,
NIVÅ 163, NIVÅ 261,
NIVÅ 263, NIVÅ 263,
NIVÅ 361, NIVÅ 362
NIVÅ 363

För att underlätta skvalptesterna är nivåerna hos givarna S61, S62 och S63 satta som konstanter. Första siffran anger tubnr

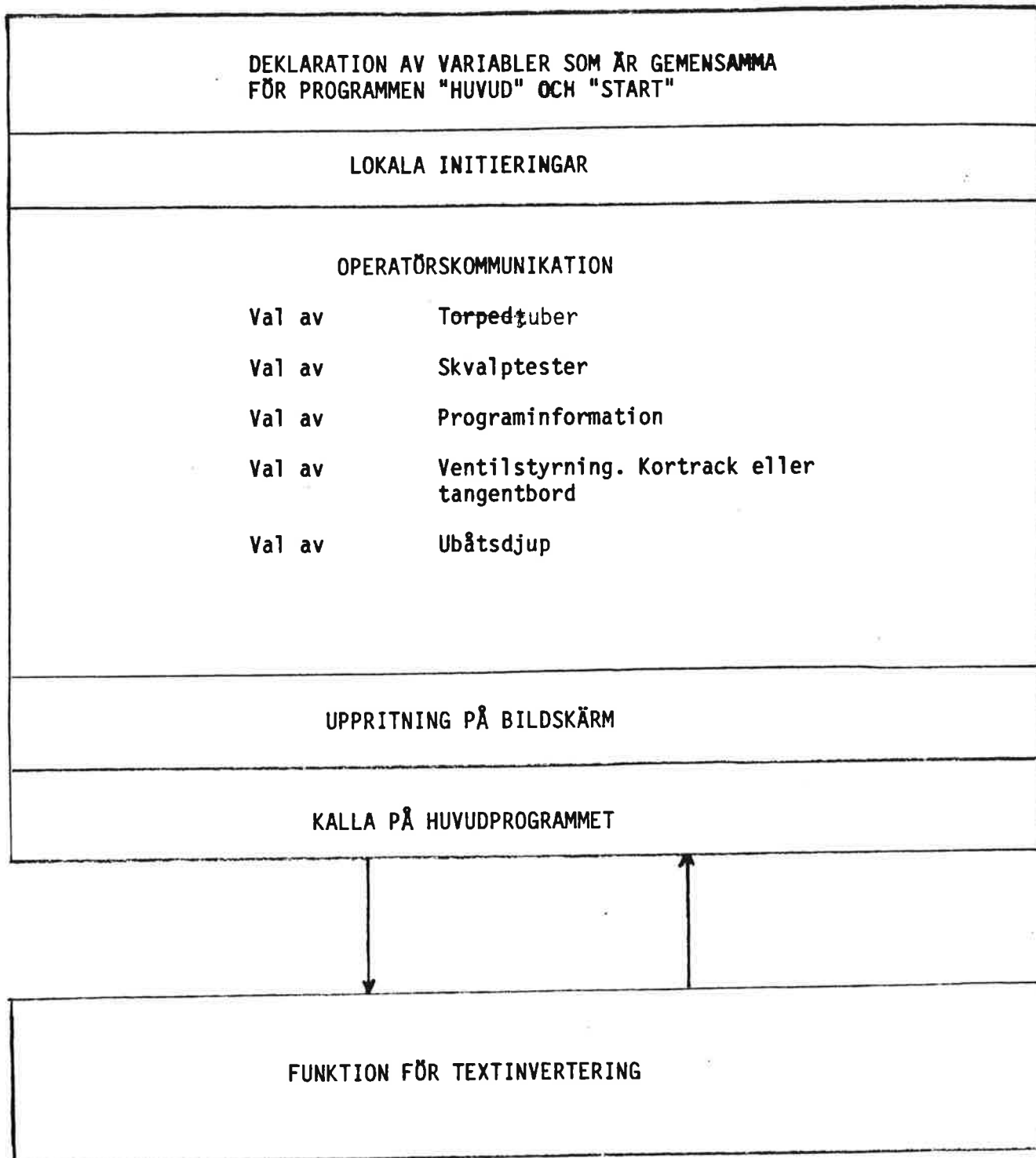
GIVARKRAFT

Flagga som anger att givarna får spänningsmatning



* Inom ett block innebär att det finns en mer detaljerad beskrivning

A.2 PROGRAMMET UPPSTART



A.3 INITIERING AV HUVUDPROGRAM

SÄTT RAD (13..17) I GRAFISK MOD

SÄTT GIVARNAS STARTVÄRDEN

SÄTT KONSTANTER FÖR NIVAGIVARE

SÄTT DE INTERVALL (I SEKUNDER) SOM TIDS-
SUBROUTINERNA SKALL VÄNTA MELLAN VARJE ANROP

ANGE STARTRAD I FÖRDRÖJNINGSMATRISERNA

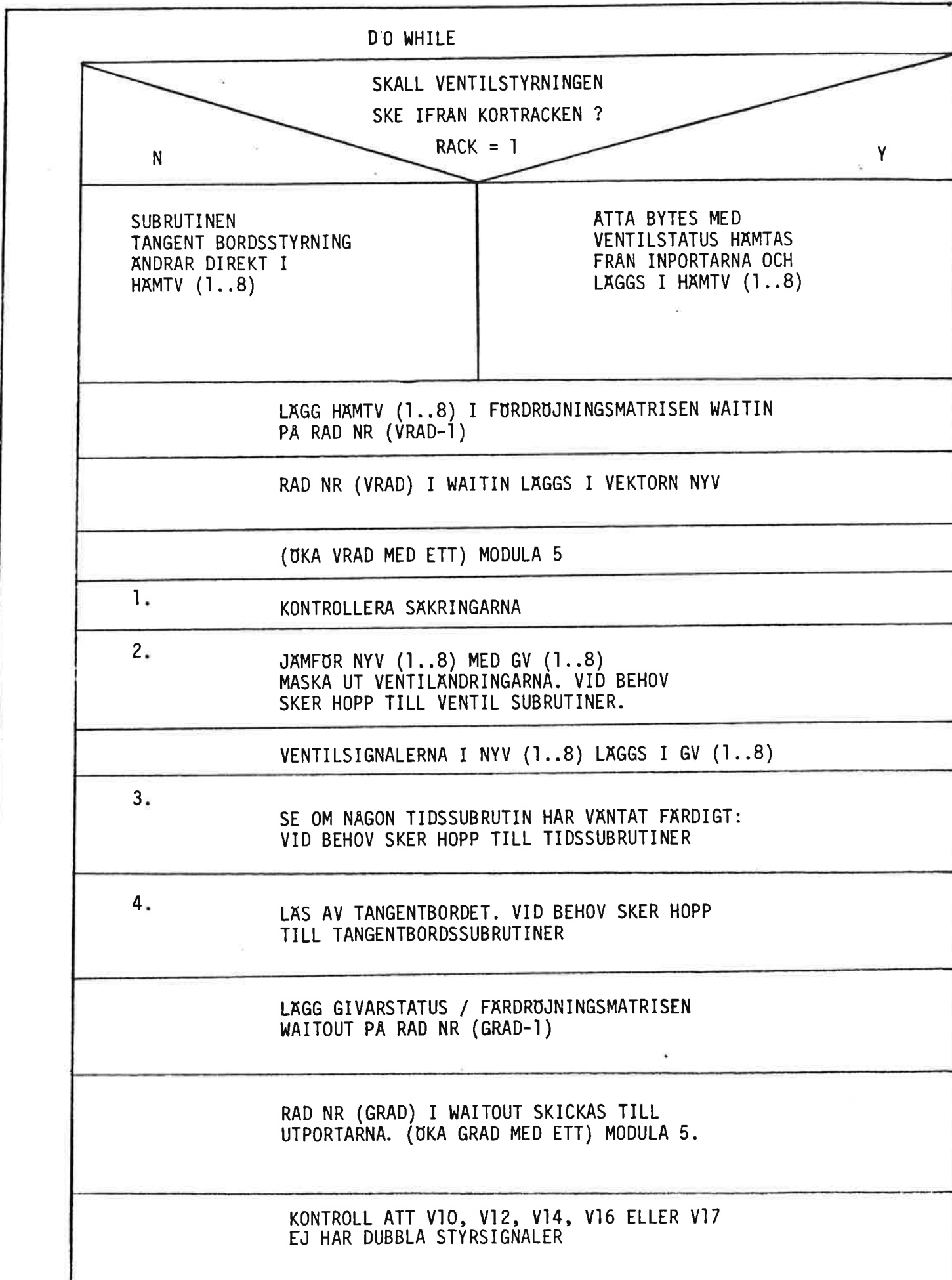
SÄTT TUBERNAS GRAFISKA POSITIONER

SÄTT STARTNIVA I TUBER

SÄTT FLAGGA SOM INDIKERAR ATT GIVARNA HAR
SPÄNNINGSMÄTNING

OBS! ALLA VARIABLER SÄTTES AUTOMATISKT TILL NOLL NÄR PROGRAMMET
STARTAS. DE VARIABLER SOM SKALL HA STARTVÄRDET NOLL ÄR
DÄRFÖR INTE MED I INITIERINGEN. DETTA FÖR ATT SPARA MINNES-
UTRYMME.

A 4. HUVUDLOOP (1000 - 4250)



A. 4.1 KONTROLL AV SÄKRINGAR (1160 - 1190)

N	SÄKRING "114" TRASIG S114 = 0?
	VENTILSIGNALERNA TILL TUB 1 + V12 + V13 SÄTTTS TILL NOLL
N	SÄKRING "115" TRASIG S115 = 0
	VENTILSIGNALERNA TILL TUB 2 SÄTTTS TILL NOLL
N	SÄKRING "116" TRASIG? S116 = 0
	VENTILSIGNALERNA TILL TUB 3 SÄTTTS TILL NOLL
N	SÄKRING "117" TRASIG? S117 = 0
	"SPÄNNINGSMATNINGEN TILL GIVARNA ÄR BORTA NOLLSTÄLL SAMTLIGA GIVARE" GIVARKRAFT = 0
N	SÄKRING "113" TRASIG? S113 = 0
	SAMTLIGA VENTILSIGNALER SÄTTTS TILL NOLL

A. 4.2.1 JÄMFÖR NYV(1) OCH GV(1) (2500 - 3010)

AR ALLA BITAR LIKA? NYV(1) = GV(1)		Y
N		
TUBNR = 1 DIFF = NYV(1) XOR GV(1)		
SKILJER SIG LÄGSTA BITEN DIFF AND 1 = 1		N
Y	GV(1) AND 1 = 1	Y
N		
GOSUB 500 "ÖPPNA V1"	GOSUB 5200 "STÅNG V1"	
DIFF AND 2 = 2		N
Y	GV(1) AND 2 = 2	Y
N		
GOSUB 5400 "ÖPPNA V3"	GOSUB 5550 "STÅNG V3"	
DIFF AND 4 = 4		N
Y	GV(1) AND 4 = 4	Y
N		
GOSUB 5600 "ÖPPNA V5"	GOSUB 6000 "STÅNG V5"	
DIFF AND 8 = 8		N
Y	GV(1) AND 8 = 8	Y
N		
GOSUB 6150 "ÖPPNA V7"	GOSUB 6300 "STÅNG V7"	
DIFF AND 16 = 16		N
Y	GV(1) AND 16 = 16	Y
N		
GOSUB 6600 "ÖPPNA (ÖPPNING AV V10)"	GOSUB 6750 "STÅNG (ÖPPNING AV V10)"	
DIFF AND 32 = 32		N
Y	GV(1) AND 32 = 32	Y
N		
GOSUB 6800 "ÖPPNA (STÅNGNING AV V10)"	GOSUB 6900 "STÅNG (STÅNGNING AV V10)"	
DIFF AND 64 = 64		N
Y	GV(1) AND 64 = 64	Y
N		
GOSUB 6950 "ÖPPNA V11"	GOSUB 7000 "STÅNG V11"	
DIFF AND 128 = 128		N
Y	GV(1) AND 128 = 128	Y
N		
GOSUB 7500 "ÖPPNA (ÖPPNING AV V14)"	GOSUB 7600 "STÅNG (ÖPPNING AV V14)"	

A. 4.2.2 JÄMFÖR NYV(2) OCH GV(2) (2500 - 3010)

ÄR ALLA BITAR LIKA NYV(2) = GV(2)		Y
N		
TUBNR = 1	DIFF = NYV(2) XOR GV(2)	
Y	SKILJER SIG LÄGSTA BITEN DIFF AND 1 = 1	N
N	GV(2) AND 1 = 1	Y
GOSUB 7650 "ÖPPNA (STÄNGNING AV V14)"	GOSUB 7700 "STÄNG (STÄNGNING AV V14)"	
Y	DIFF AND 2 = 2	N
N	GV(2) AND 2 = 2	Y
GOSUB 7750 "ÖPPNA V 15"	GOSUB 7800 "STÄNG V15"	
Y	DIFF AND 4 = 4	N
N	GV(2) AND 4 = 4	Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"	
Y	DIFF AND 8 = 8	N
N	GV(2) AND 8 = 8	Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"	
Y	DIFF AND 16 = 16	N
N	GV(2) AND 16 = 16	Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"	
Y	DIFF AND 32 = 32	N
N	GV(2) AND 32 = 32	Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"	
Y	DIFF AND 64 = 64	N
N	GV(2) AND 64 = 64	Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"	
Y	DIFF AND 128 = 128	N
N	GV(2) AND 128 = 128	Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"	

A. 4.2.3 JÄMFÖR NYV(3) OCH GV(3) (2500 - 3010)

N		ÄR ALLA BITAR LIKA NYV(3) = GV(3)		Y
TUBNR = 2		DIFF = NYV(3) XOR GV(3)		
Y	SKILJER SIG LÄGSTA BITEN DIFF AND 1=1			N
N	GV(3) AND 1 = 1		Y	
GOSUB 5000 "ÖPPNA V1"		GOSUB 5200 "STÄNG V1"		
Y	DIFF AND 2 = 2			N
N	GV(3) AND 2 = 2		Y	
GOSUB 5400 "ÖPPNA V3"		GOSUB 5550 "STÄNG V3"		
Y	DIFF AND 4 = 4			N
N	GV(3) AND 4 = 4		Y	
GOSUB 5600 "ÖPPNA V5"		GOSUB 6000 "STÄNG V5"		
Y	DIFF AND 8 = 8			N
N	GV(3) AND 8 = 8		Y	
GOSUB 6150 "ÖPPNA V7"		GOSUB 6300 "STÄNG V7"		
Y	DIFF AND 16=16			N
N	GV(3) AND 16=16		Y	
GOSUB 6600 "ÖPPNA (ÖPPNING AV V10)"		GOSUB 6750 "STÄNG (ÖPPNING AV V10)"		
Y	DIFF AND 32=32			N
N	GV(3) AND 32=32		Y	
GOSUB 6800 "ÖPPNA (STÄNGNING AV V10)"		GOSUB 6900 "STÄNG (STÄNGNING AV V10)"		
Y	DIFF AND 64=64			N
N	GV(3) AND 64=64		Y	
GOSUB 6950 "ÖPPNA V11"		GOSUB 700 "STÄNG V11"		
Y	DIFF AND 128=128			N
N	GV(3) AND 128=128		Y	
GOSUB 7500 "ÖPPNA (ÖPPNING AV V14)"		GOSUB 7600 "STÄNG (ÖPPNING AV V14)"		

A. 4.2.4 JÄMFÖR NYV(4) OCH GV(4) (2500 - 3010)

		ÄR ALLA BITAR LIKA NYV(4) = GV(4)	Y
N	TUBNR = 2	DIFF = NYV(4) XOR GV(4)	
Y	SKILJER SIG LÅGSTA BITEN DIFF AND 1=1		N
N	GV(4) AND 1 = 1	Y	
GOSUB 7650 "ÖPPNA (STÄNGNING AV V14)"		GOSUB 7700 "STÄNG (STÄNGNING AV V14)"	
Y	DIFF AND 2 = 2		N
N	GV(4) AND 2 = 2	Y	
GOSUB 7750 "ÖPPNA V15"		GOSUB 7800 "STÄNG V15"	
Y	DIFF AND 4 = 4		N
N	GV(4) AND 4 = 4	Y	
GOSUB 14300 "TOM SUBRUTIN"		GOSUB 14300 "TOM SUBRUTIN"	
Y	DIFF AND 8 = 8		N
N	GV(4) AND 8 = 8	Y	
GOSUB 14300 "TOM SUBRUTIN"		GOSUB 14300 "TOM SUBRUTIN"	
Y	DIFF AND 16 = 16		N
N	GV(4) AND 16 = 16	Y	
GOSUB 14300 "TOM SUBRUTIN"		GOSUB 14300 "TOM SUBRUTIN"	
Y	DIFF AND 32 = 32		N
Y	GV(4) AND 32 = 32	Y	
GOSUB 14300 "TOM SUBRUTIN"		GOSUB 14300 "TOM SUBRUTIN"	
Y	DIFF AND 64 = 64		N
N	GV(4) AND 64 = 64	Y	
GOSUB 14300 "TOM SUBRUTIN"		GOSUB 14300 "TOM SUBRUTIN"	
Y	DIFF AND 128=128		N
N	GV(4) AND 128=128	Y	
GOSUB 14300 "TOM SUBRUTIN"		GOSUB 14300 "TOM SUBRUTIN"	

A.4.2.5. JÄMFÖR NYV(5) OCH GV(5) (2500-3010)

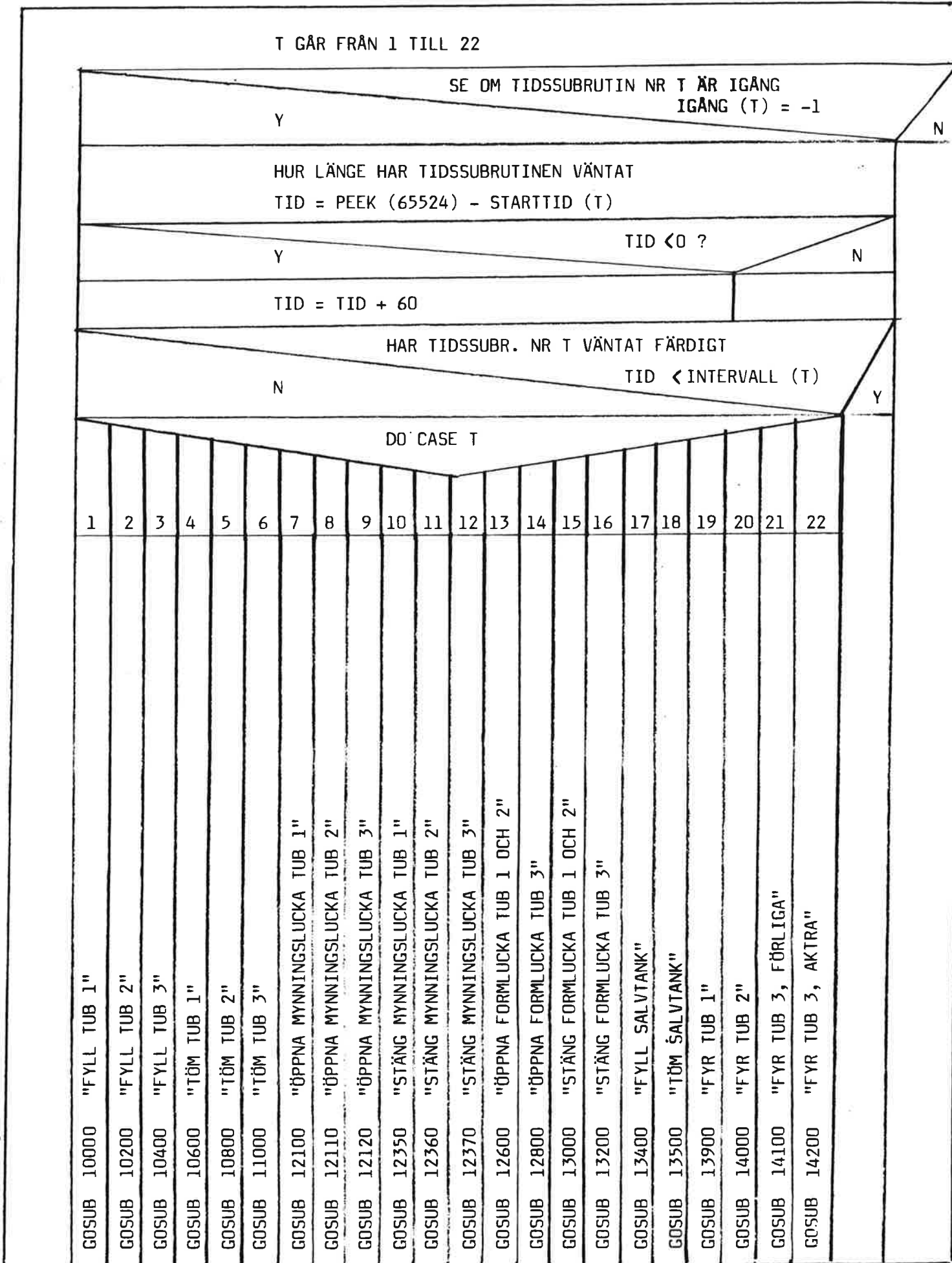
		ÄR ALLA BITAR LIKA? NYV(5) = GV(5)	
N			Y
TUBNR = 3	DIFF = NYV(5) XOR GV(5)	SKILJER SIG LAGSTA BITEN	
Y		DIFF AND 1=1	
N	GV(5) AND 1 = 1		Y
GOSUB 5000 "ÖPPNA VI"	GOSUB 5200 "STÄNG VI"		
Y		DIFF AND 2=2	N
N	GV(5) AND 2 = 2		Y
GOSUB 5400 "ÖPPNA V3"	GOSUB 5550 STÄNG V3"		
Y		DIFF AND 4=4	N
N	GV(5) and 4=4		Y
GOSUB 5600 "ÖPPNA V5"	GOSUB 6000 "STÄNG V5"		
Y		DIFF AND 8=8	N
N	GV(5) AND 8=8		Y
GOSUB 6150 "ÖPPNA V7"	GOSUB 6300 "STÄNG V7"		
Y		DIFF AND 16=16	N
N	GV(5) AND 16=16		Y
GOSUB 6600 "ÖPPNA (ÖPPNING AV VID)"	GOSUB 6750 "STÄNG (ÖPPNING AV VID)"		
Y		DIFF AND 32=32	N
N	GV(5) AND 32=32		Y
GOSUB 6800 "ÖPPNA (STÄNGNING AV VID)"	GOSUB 6900 "STÄNG (STÄNGNING AV VID)"		
Y		DIFF AND 64=64	N
N	GV(5) AND 64=64		Y
GOSUB 6950 "ÖPPNA V11 "	GOSUB 7000 "STÄNG V11"		
Y		DIFF AND 128=128	N
N	GV(5) AND 128=128		Y
GOSUB 7050 "ÖPPNA (ÖPPNING V12)"	GOSUB 7150 "STÄNG (ÖPPNING V12)"		

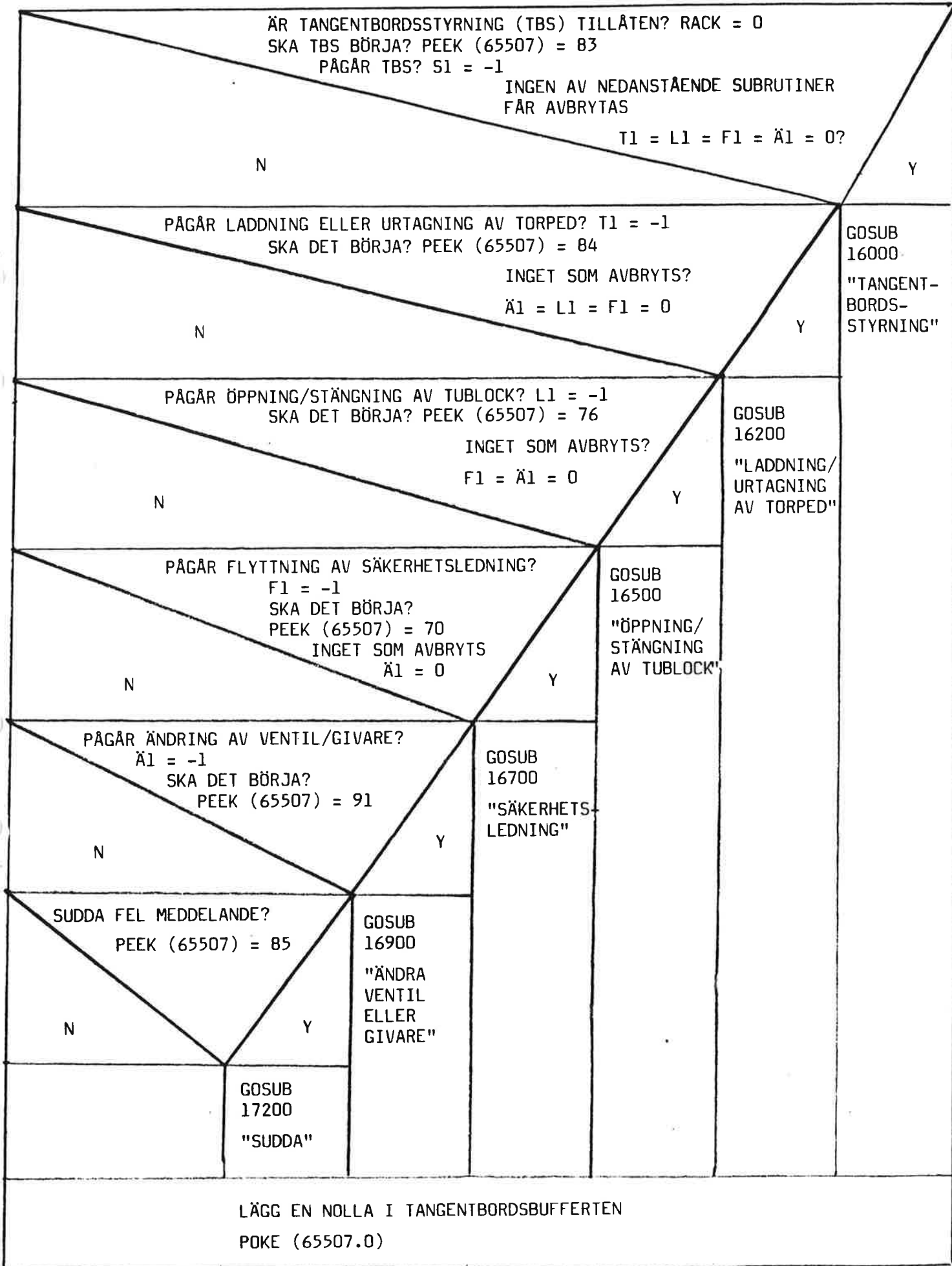
A.4.2.6. JÄMFÖR NYV(6) OCH GV(6) (2500-3010)

		ÄR ALLA BITAR LIKA? NYV(6) = GV(6)	
N			Y
TUBNR = 3	DIFF = NYV(6) XOR GV(6)		
Y	SKILJER SIG LÄGSTA BITEN DIFF AND 1=1		N
N	GV(6) AND 1 = 1	Y	
GOSUB 7200 "ÖPPNA (STÄNGNING AV V12)"	GOSUB 7350 "STÄNG (STÄNGNING AV V12)"		
Y	DIFF AND 2 = 2		N
N	GV(6) AND 2 = 2	Y	
GOSUB 7400 "ÖPPNA V13"	GOSUB 7450 "STÄNG V13"		
Y	DIFF AND 4 = 4		N
N	GV(6) AND 4 = 4	Y	
GOSUB 7850 "ÖPPNA (ÖPPNING AV V16)"	GOSUB 7950 "STÄNG (ÖPPNING AV V16)"		
Y	DIFF AND 8 = 8		N
N	GV(6) AND 8 = 8	Y	
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		
Y	DIFF AND 16 = 16		N
N	GV(6) AND 16 = 16	Y	
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		
Y	DIFF AND 32 = 32		N
N	GV(6) AND 32 = 32	Y	
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		
Y	DIFF AND 64 = 64		N
N	GV(6) AND 64 = 64	Y	
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		
Y	DIFF AND 128 = 128		N
N	GV(6) AND 128 = 128	Y	
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		

N		ÄR ALLA BITAR LIKA NYV(7) = GV(7)	Y
TUBNR = 3		DIFF = NYV(7) XOR GV(7)	
Y		SKILJER SEJ LÄGSTA BITEN DIFF AND 1 = 1	N
N		GV(7) AND 1 = 1	Y
GOSUB 8000 "ÖPPNA (STÄNGNING AV V16)"	GOSUB 8050 "STÄNG (STÄNGNING AV V16)"		
Y		DIFF AND 2 = 2	N
N		GV(7) AND 2 = 2	Y
GOSUB 8100 "ÖPPNA (ÖPPNING AV V17)"	GOSUB 8250 "STÄNG (ÖPPNING AV V17)"		
Y		DIFF AND 4 = 4	N
N		GV(7) AND 4 = 4	Y
GOSUB 8300 "ÖPPNA (STÄNGNING AV V17)"	GOSUB 8350 "STÄNG (STÄNGNING AV V17)"		
Y		DIFF AND 8 = 8	N
N		GV(7) AND 8 = 8	Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		
Y		DIFF AND 16 = 16	N
N		GV(7) AND 16 = 16	Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		
Y		DIFF AND 32 = 32	N
N		GV(7) AND 32 = 32	Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		
Y		DIFF AND 64 = 64	N
N		GV(7) AND 64 = 64	Y
TUBNR = 1	TUBNR = 1		
GOSUB 6400 "ÖPPNA V9"	GOSUB 6500 "STÄNG V9"		
Y		DIFF AND 128 = 128	N
N		GV(7) AND 128 = 128	Y
TUBNR = 1	TUBNR = 1		
"ÖPPNA (ÖPPNING AV V12)"	"STÄNG (ÖPPNING AV V12)"		

		ÄR ALLA BITAR LIKA NYV(8) = GV(8)	
N			Y
TUBNR = 2	DIFF = NYV(8) XOR GV(8)		
Y	SKILJER SEJ LÄGSTA BITEN DIFF AND 1 = 1		N
N	GV(8) AND 1 = 1		Y
GOSUB 7200 "ÖPPNA (STÄNGNING AV V12)"	GOSUB 7350 "STÄNG (STÄNGNING AV V12)"		
Y	DIFF AND 2 = 2		N
N	GV(8) AND 2 = 2		Y
GOSUB 7400 "ÖPPNA V13"	GOSUB 7450 "STÄNG V13"		
Y	DIFF AND 4 = 4		N
N	GV(8) AND 4 = 4		Y
GOSUB 8400 "ÖPPNA V18"	GOSUB 8400 "STÄNG V18"		
Y	DIFF AND 8 = 8		N
N	GV(8) AND 8 = 8		Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		
Y	DIFF AND 16 = 16		N
N	GV(8) AND 16 = 16		Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		
Y	DIFF AND 32 = 32		N
N	GV(8) AND 32 = 32		Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		
Y	DIFF AND 64 = 64		N
N	GV(8) AND 64 = 64		Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		
Y	DIFF AND 128 = 128		N
N	GV(8) AND 128 = 128		Y
GOSUB 14300 "TOM SUBRUTIN"	GOSUB 14300 "TOM SUBRUTIN"		





B. FYSISKT SYSTEM**B.1 IN/UT-KORT**

I kortracken sitter det 9 st kretskort av "Databoard-typ". Två av dessa har 32 st reläer och används för att omvandla ventilsignalerna från referenssystemet till en för ABC-802 användbar representation. Kortet får 32 st olika spänningar från referenssystemet och lämnar vid förfrågan 4 st 8-bitars tal till ABC-datorn.

De sju övriga korten mottar vardera två 8-bitars tal från ABC-802 och omvandlar dessa till 16 olika spänningar som kan avläsas av referenssystemet. Det är givarnas status som skickas på detta sätt.

På nästföljande två sidor följer scheman som visar vilken bit på korten som hör ihop med respektive ventil/givare.

Därefter följer "Databoards" egen beskrivning av in/ut-korten. Beskrivningen ger ökad förståelse för bithantering och kan vara användbar vid framtida ändringar i programmet.

Mottagare

IN-KORT

Kort 8

Byte 1 (tub 1)

7	6	5	4	3	2	1	0
V14Ö	V11	V10S	V10Ö	V7	V5	V3	V1

Byte 2 (tub 1)

7	6	5	4	3	2	1	0
1	1	1	1	1	0	V15	V14S

Byte 3 (Tub 2)

7	6	5	4	3	2	1	0
V14Ö	V11	V10S	V10Ö	V7	V5	V3	V1

Byte 4 (tub 2)

7	6	5	4	3	2	1	0
1	1	1	1	1	0	V15	V14S

Kort 9

Byte 5 (tub 3)

7	6	5	4	3	2	1	0
V12Ö	V11	V10S	V10Ö	V7	V5	V3	V1

Byte 6 (tub 3)

7	6	5	4	3	2	1	0
1	1	1	1	0	V16Ö	V13	V12S

Byte 7 (gemensam)

7	6	5	4	3	2	1	0
V12Ö	V9	1	1	1	V17S	V17Ö	V16S

Byte 8 (gemensam)

7	6	5	4	3	2	1	0
1	1	1	1	1	18	13	12S

Mottagare

UT-KORT

Kort 1	<u>Byte 1 (tub 1)</u>	<u>Byte 2 (tub 1)</u>
	63 62 61 60 58 57 52 51	65 - 75 74 73 72 71 70
Kort 2	<u>Byte 3 (tub 2)</u>	<u>Byte 4 (tub 2)</u>
	63 62 61 60 58 57 52 51	65 - 75 74 73 72 71 70
Kort 3	<u>Byte 5 (tub 1)</u>	<u>Byte 6 (tub 2)</u>
	97 96 95 94 93 92 91 90	97 96 95 94 93 92 91 90
Kort 4	<u>Byte 7 (tub 3)</u>	<u>Byte 8 (tub 3)</u>
	63 62 61 60 58 57 - -	56 55 54 53 73 72 71 70
Kort 5	<u>Byte 9 (tub 3)</u>	<u>Byte 10 (tub 3)</u>
	65 - 85 84 83 82 81 80	97 96 95 94 93 92 91 90
Kort 6	<u>Byte 11 (gemensamma)</u>	<u>Byte 12 (gemensamma)</u>
	105 104 103 102 101 100 99 98	- - 119 118 109 108 107 106
Kort 7	<u>Byte 13 (gemensamma)</u>	<u>Byte 14 (gemensamma)</u>
	- 117 116 115 114 113 112 111	- 88 87 86 83 82 81 80

OCT 81 1 6

APPENDIX B
PM 3948
Sid 4

CONTENTS

- 1. Description
- 1. Applications
- 2. Technical data
- 3. Installation
- 4. Commands
- 5. Test-examples
- 6. Block-diagram
- 6. I/O-connector

INNEHÅLL

- 1. Beskrivning
- 1. Användningsområde
- 2. Tekniska data
- 3. Installation
- 4. Kommandon
- 5. Test-exempel
- 6. Block-schema
- 6. I/O-kontakt

DESCRIPTION

- 32 input lines, in four groups of 8 inputs each. Each group has a control input to select latching or transparent inputs.
- 4 interrupt inputs for general interrupt or as ordinary digital inputs. The interrupt inputs may be connected to the control inputs for latching and interrupt at the same time. By means of jumpers on the connector it is possible to select interrupt on rising or falling edge. Individual interrupts can be enabled or disabled by program.
- 2 output lines for general use or for handshake purpose in handling the incoming interrupt signal.

BESKRIVNING

- 32 ingångar i fyra 8 bitars grupper. Varje grupp har en kontrollingång för val av lagring i vippor eller direkt avläsning av signalerna.
- 4 avbrottsingångar för avbrotts hantering eller att avläsas som vanliga digitala ingångar. Avbrottsingångarna kan kopplas samman med kontrollingångarna för lagring av signaler samtidigt som avbrott genereras. Varje avbrottsingång kan byglas för att generera avbrott på positiv eller negativ flank. Programmet bestämmer vilka avbrottsingångar som ska generera interrupt.
- 2 utgångar att användas för handskakning i samband med avbrott eller som vanliga digitala utsignaler.

APPLICATIONS

- Systems which requires interrupt handling.
- System which needs many inputs and few outputs.

ANVÄNDNINGSSOMRÅDEN

- I system som kräver avbrotts hantering.
- I system som behöver många ingångar och få utgångar.

INSTALLATION

INSTALLATION

APPENDIX B
PM 3948
Sid 5

1. Select address.

All cards in the system must have different addresses. Suggestion: address 5, see figures.

2. When using interrupt.

If strobe and interrupt shall be generated at the same time, connect strobe inputs 3B, 8B, 13B and 18B to interrupt inputs 26B, 27B, 28B and 29B. Interrupt is generated and inputs latched when external interrupt signal turns from '1' to '0'. The signal must be low until the computer read the inputs. For interrupt on rising edge ('0' to '1') connect corresponding INTFAS to GND.

3. Connection of strobe 0-3.

For the groups which only uses transparent inputs, connect corresponding strobe-input to one free INTFAS input (+5V pullup-resistor). Otherwise connect strobe-inputs to external circuits. The inputs will be latched when strobe falls.

4. Connect external unit to the I/O-connector (the connector nearest the LED).

5. Insertion.

SWITCH THE POWER OFF
Turn the I/O-connector outwards and the component side to the right.
Put the card in the I/O-side.

6. Check the address plug.

By writing OUT 1,A (BASIC, A = address of code plug), the LED shall turn on. Run the following program:
10 OUT 1,5 : GOTO 10
to turn the LED on (address 5).

1. Välj adress.

Alla kort i systemet ska ha olika adresser. Förslag adress 5, se figur.

2. Vid användning av avbrott.

Om strobe och avbrott ska genereras samtidigt, anslut strobe ingångarna 3B, 8B, 13B och 18B till avbrotts ingångarna 26B, 27B, 28B och 29B. Avbrott genereras och indata lagras när yttre avbrottssignal går från '1' till '0'. Signalen ska hållas låg tills datorn läst av ingångarna. För avbrott på positiv flank ('0' to '1') motsvarande INTFAS till jord.

3. Anslutning av strobe 0-3.

För de grupper som används endast till direktavläsning ansluts motsvarande strobe-ingång till en ledig INTFAS ingång (+5V genom motstånd). I annat fall anslut strobe-ingång till externa kretsar. Ingångarna kommer att läggas i vippor när strobe går låg.

4. Anslut yttre enhet till I/O-kontakten (kontakten närmast lysdioden).

5. Insättning.

SLÅ AV SPÄNNINGEN
Vänd I/O-kontakten utåt och komponent sidan åt höger.
Placera kortet i I/O-delen.

6. Kontrollera adresspluggen.

Genom att skriva OUT 1,A (BASIC, A anger adress), ska lysdioden tändas. Kör följande program:
10 OUT 1,5 : GOTO 10
för att tända lysdioden (adress 5).

=====

TECHNICAL DATA

TEKNISKA DATA

Power Supply
Spänningsmatning

+ 5V +/- 5% 600mA

Bus connection

I/O-side. Includes signal CSB* for busexpansion, see system-manual.

Anslutning till bussen

I/O sidan. Bussanslutningen inkluderar signalen CSB* för busexpansion, se system manual.

Connector
KontaktidonB 64 pin Standard Europe connector.
(DIN 41612) on both I/O- and bus-side.Size
Storlek

Standard Europe card, 100 x 160 mm.

Interfacing

32 TTL-inputs in four groups of 8 inputs each. Load = 1 TTL.
Each group is controlled by a strobe input.

4 strobe-inputs. Load = 1 TTL.
Strobe = "1" transparent.
= "0" latching.

4 interrupt-lines. Load = 1 TTL.
2 outputs. Driving capacity = 10 TTL.

Anpassning

32 TTL-ingångar i fyra grupper med vardera 8 ingångar.
Belastning = 1 TTL.
Varje grupp kontrolleras med strobe ingång.

4 strobe-ingångar.
Belastning = 1 TTL.
Strobe = "1" direkt avläsning.
= "0" lagring i vippor.

4 avbrottsingångar. Belastning = 1 TTL
2 TTL-utgångar.
Belastningsbarhet = 10 TTL.

Interrupt control

By means of jumpers on cable connector selectabel to rising or falling edge.
INTFAS = "1" falling edge.
INTFAS = "0" rising edge.

Generering av avbrott

Valbart avbrott vid positiv eller negativ flank, bestäms med bygling i anslutningskablagerets kontaktdon.
INTFAS = "1" negativ flank.
INTFAS = "0" positiv flank.

COMMANDS

KOMMANDON
FUNCTION (EXAMPLE IN BASIC)

Signal RST
 ASSEMB. INP 7
 FORTRAN INPUT(7)
 PASCAL INP(7)
 BASIC INP(7)
 Example 10 A=INP(7)

Reset the I/O-system. All programs should have this command at the beginning.

Nollställer alla I/O-kort. Satsen bör finnas med i början på alla program.

Signal CS
 ASSEMBL OUT 1
 FORTRAN OUTPUT(1)=A
 PASCAL OUT(1,A)
 BASIC OUT 1,A
 Example 20 OUT 1,5

A=0..63
 Select card with address A. The LED on the card is turned on.

Väljer kort med adress A. När satsen utförs tänds lysdioden på kortet.

Signal OUT
 ASSEMBL OUT 0
 FORTRAN OUTPUT(0)=X+Y
 PASCAL OUT(0,X+Y)
 BASIC OUT 0,X+Y
 Example 30 OUT 0,2+8

X=0..3 (bit 0,1), Y=0,4,8,12 (bit 2,3)
 X selects inputs group.
 Y controls outputs A and B. Bit 2 OUTP A, bit 3 OUTP B.

X väljer ingångsgrupp.
 Y styr utgångarna A och B. Bit 2 utgång A bit 3 utgång B.

Signal INP
 ASSEMBL INP 0
 FORTRAN INPUT(0)
 PASCAL INP(0)
 BASIC INP(0)
 Example 40 B=INP(0)

Reads 8 bit of data from selected group. See command OUT 0

Läser 8 bitar från utvald grupp. Se kommando OUT 0.

Signal STAT
 ASSEMB. INP 1
 FORTRAN INPUT(1)
 PASCAL INP(1)
 BASIC INP(1)
 Example 50 C=INP(1)

Used to acknowledge interrupt, reset INT. Reads the four interrupt inputs. Bit 0 for interrupt-input 0, etc. 3 for interrupt-input 3. Bit-state "1" = passive interrupt-signal. "0" = active interrupt-signal.

Bekräftar avbrott (nollställer INT).
 Läser de fyra avbrottsingångarna.
 Bit 0 för avbrotts-ingång 0, etc.
 3 för avbrotts-ingång 3.
 Bit-tillstånd "1" = passiv ingång.
 "0" = aktiv ingång.

Signal C4
 ASSEMB. OUT 5
 FORTRAN OUTPUT(5)=X
 PASCAL OUT(5,X)
 BASIC OUT 5,X
 Example 60 OUT 5,2

X=0..15 (bit 0,1,2,3)
 Selective interrupt enable/disable contr. Bit 0 controls interrupt on INT 0, etc. 3 controls interrupt on INT 3. Bit-state "1" = enable. "0" = disable.

Val av avbrottsingångar
 Bit 0 kontrollerar INT 0, etc.
 3 kontrollerar INT 3.
 Bit-tillstånd "1" = avbrott tillåtet.
 "0" = avbrott bortkopplat.

Reg A contains suitable parameter in the assembler commands.
 Reg A innehåller aktuell parameter i assembler kommandon.

TEST-EXAMPLES

TEST-EXEMPEL

The programs are written in BASIC.
The first test-example shows how to read a input-group.
The second shows how to control the two outputs.
The third shows how to read the interrupt-signals.
In test-example 1 and 3, reading of inputs occurs every time a key is pressed.

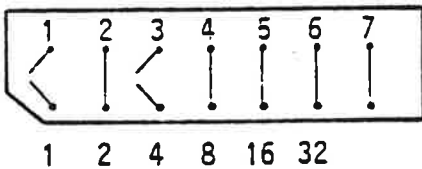
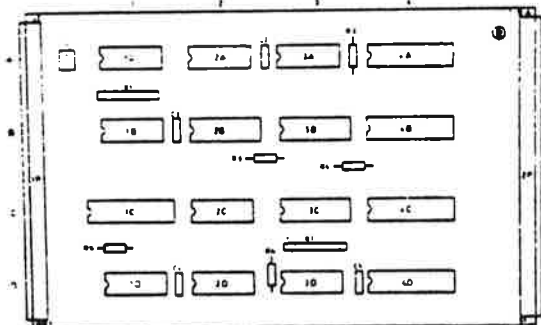
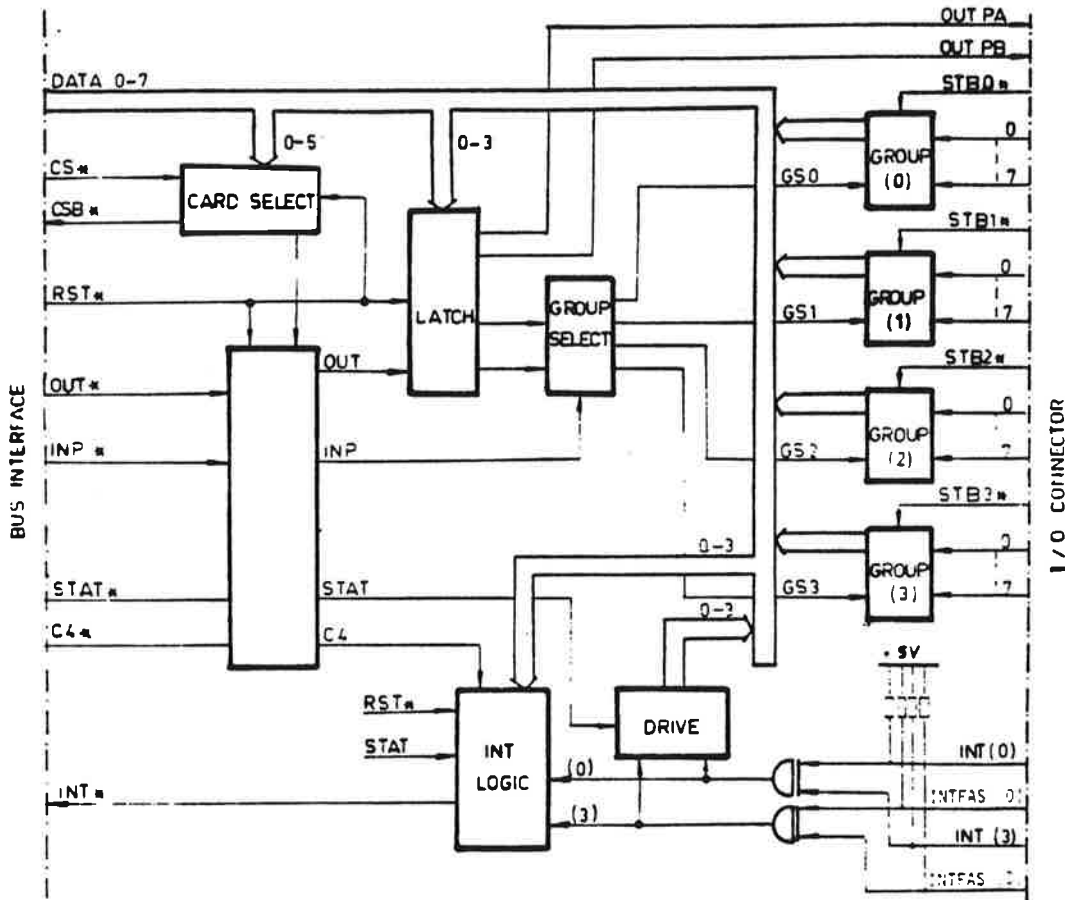
Programmen är skrivna i BASIC.
Test-exempel 1 visar hur man läser en indatagrupp 8 bitar.
Med exempel 2 kontrolleras de två utgångarna.
Exempel 3 läser av avbrottsingångarna.
I exempel 1 och 3 sker läsning av ingångar varje gång en tangent trycks ned.

```
10 REM IN4085
30 OUT 1,9
50 PRINT "WHICH GROUP (0-3)";:INPUT G
60 IF G<0 OR G>3 THEN 50
70 OUT 0,G
80 PRINT "INPUTS DECIMAL =";INP(0)
90 FOR IX=0 TO 7
100 PRINT "BIT";IX;(INP(0) AND 2%xxIX)/2%xxIX
110 NEXT IX
120 GET A$
130 GOTO 50
```

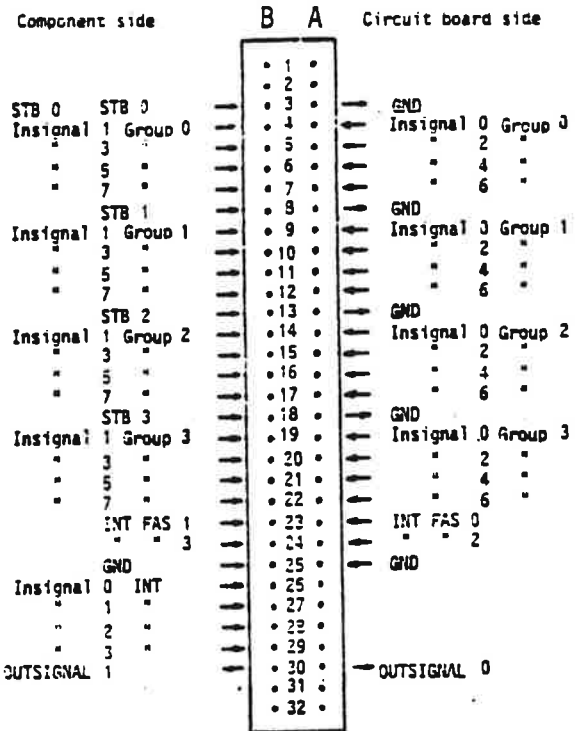
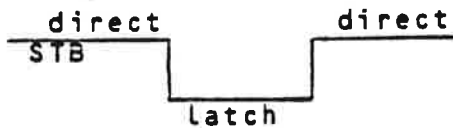
```
10 REM OUT4085
30 OUT 1,9
50 PRINT "OUTP A =";A%
60 PRINT "OUTP B =";B%
70 PRINT "OUTP A =";:INPUT A%
80 IF A%<0 OR A%>1 THEN 70
90 PRINT "OUTP B =";:INPUT B%
100 IF B%<0 OR B%>1 THEN 90
110 OUT 0,4xA%+8xB%
120 GOTO 50
```

```
10 REM INTERRUPT-SIGNALS
30 OUT 1,9
50 PRINT "INPUTS DECIMAL =";INP(1)
60 FOR IX=0 TO 3
70 PRINT "BIT";IX;(INP(1) AND 2%xxIX)/2%xxIX
80 NEXT IX
90 GET A$
100 GOTO 50
```

=====



Cut jumpers 1 and 3-
gives adress 1+4=5.
Position of code-plug:1A
Jumper 7 is not used.



INTFAS selects if interrupt is triggered on rising or falling signal edge on INT-input.
Connect to 0V (in connector) for rising edge trigg. INTFAS is internally pulled up for falling edge trigg.

INTFAS väljer om interrupt ska ske på positiv eller negativ flank på INT-input signalen.
Bygla i kontaktdonet till 0V för positiv trigg. INTFAS är via ett motstånd internt ansluten till +5V för negativ trigg.

DATABOARD 4680

16 RELAY OUTPUTS

4103

=====
OKT 81 1 6

CONTENTS

- 1. Description
- 1. Applications
- 2. Technical data
- 3. Installation
- 4. Commands
- 5. Test-examples
- 6. Block-diagram
- 6. I/O-connector

INNEHÅLL

- 1. Beskrivning
- 1. Användningsområde
- 2. Tekniska data
- 3. Installation
- 4. Kommandon
- 5. Test-exempel
- 6. Block-schema
- 6. I/O-kontakt

=====

DESCRIPTION

- 16 Relays with one switching contact.
- The relays are controlled in 2 groups of each 8 relays.
- One command to open all contacts.
- Selectable internal or external relay drive voltage.
- The card is program compatible with the outputs on the 4005. The card is also program and connection compatible, with respect to closing, to the card 4095.

=====

BESKRIVNING

- 16 Relä med en växlande kontakt.
- Reläerna styrs i 2 grupper om vardera 8 reläer.
- Ett kommando för att bryta alla kontakter.
- Valbart intern eller yttre drivspänning till relä.
- Kortet är programkompatibelt med utgångarna på 4005. Kortet är också program och anslutningskompatibelt, med avseende på slutning, till kort 4095.

=====

APPLICATIONS

- Control of lamps, alarms etc.
- Control of galvanic separated systems.

=====

ANVÄNDNINGSSOMRÅDEN

- Styrning av lampor, alarm etc
 - Kontroll av galvanisk skilda system.
- =====

4103

Okt 81 2 6

=====

TECHNICAL DATA

TEKNISKA DATA

Power Supply	+ 5V +/- 5%. Current 500mA when internal 5V is used for driving the relays.
Spänningsmatning	+ 5V +/- 5%. Ström 500mA med intern 5V kopplad till reläerna.
Bus connection Anslutning till bussen	I/O-side.
Connector Kontaktidon	B 64 pin Standard Europe connector (DIN 41612) on both I/O- and bus-side.
Size Storlek	Standard Europe card, 100 x 160 mm.
Relay, type RH-5	Contact voltage maximum 100V. Contact current maximum 1A. Break power maximum 20W. Supply voltage 5V. Switch 'ON'- 'OFF'-time 1 millisecond.
Relä, typ RH-5	Kontaktspänning max 100V. Kontaktström max 1A. Bryteffekt max 20W. Drivspänning 5V. Från- och till-slagstid 1 millisekund.

=====

4103

Okt 81 3 6

=====

INSTALLATION

1. Select address.
All cards in the system must have different addresses. Suggestion: address 3, see figures.
2. If external driving voltage to the relays is wanted, move jumper on position 3E as in the figure.



Connect external driving voltage +5V to 29A/B, and GND to 2A/B on the I/O-connector (2P).

3. Connect external unit to the I/O-connector (the connector nearest the LED).
4. Insertion.
SWITCH THE POWER OFF
Turn the component side to the right.
Put the card in the I/O-side.
5. Check the address plug.
The following BASIC program turns the LED on.
A=address of code plug.
A=3 if the code plug is coded as in the fig.
10 OUT 1,A : GOTO 10

INSTALLATION

1. Välj adress.
Alla kort i systemet ska ha olika adresser. Förslag adress 3, se figur.
2. Om extern drivspänning till relä önskas flytta bygling på position 3E som i figuren nedan.



Anslut yttre drivspänning 5V till stift 29A/B, jord till stift 2A/B på I/O-kontakten (2P).

3. Anslut yttre enhet till I/O-kontakten (kontakten närmast lysdioden).
4. Insättning.
BRYT SPÄNNINGEN
Vänd komponentsidan åt höger.
Placera kortet i I/O-delen.
5. Kontrollera adresspluggen.
Följande BASIC program tänds lysdioden på kortet.
A=adress på kodpluggen.
A=3 för kodplugg enligt figur.
10 OUT 1,A : GOTO 10

=====

4103

OKT 81 4 6

=====

COMMANDS

KOMMANDON

FUNCTION (EXAMPLE IN BASIC)

 Signal CS A=0..63
 ASSEMB. OUT 1 Select card with address A. The LED on the
 FORTRAN OUTPUT(1)=A card is turned on indicating selection.
 PASCAL OUT(1,A)
 BASIC OUT 1,A Väljer kort med adress A. När satsen ut-
 Example 20 OUT 1,3 förs tänds lysdioden på kortet.

Signal OUT A=0..255 (bit 0-7)
 ASSEMBL OUT 0 Byte 2 of integer A is send as data to
 FORTRAN OUTPUT(0)=A group 1, relay 0-7.
 PASCAL OUT(0,A)
 BASIC OUT 0,A Byte 2 av heltalsvariabeln A skickas som
 Example 30 OUT 0,4 data till grupp 1, relä 0-7.

Signal C1 A=0..255 (bit 0-7)
 ASSEMB. OUT 2 Byte 2 of integer A is send as data to
 FORTRAN OUTPUT(2)=A group 2, relay 8-15.
 PASCAL OUT(2,A)
 BASIC OUT 2,A Byte 2 av heltalsvariabeln A skickas som
 Example 40 OUT 2,16 data till grupp 2, relä 8-15.

Signal C3 Opens all the 16 relay contacts.
 ASSEMB. OUT 4
 FORTRAN OUTPUT(4)=0 Bryter alla 16 reläkontakterna.
 PASCAL OUT(4,0)
 BASIC OUT 4,0
 Example 50 OUT 4,0

Reg A contains suitable parameter in the assembler commands.

Reg A innehåller aktuell parameter i assembler kommandon.

=====

4103

OKT 81 5 6

=====

TEST-EXAMPLES

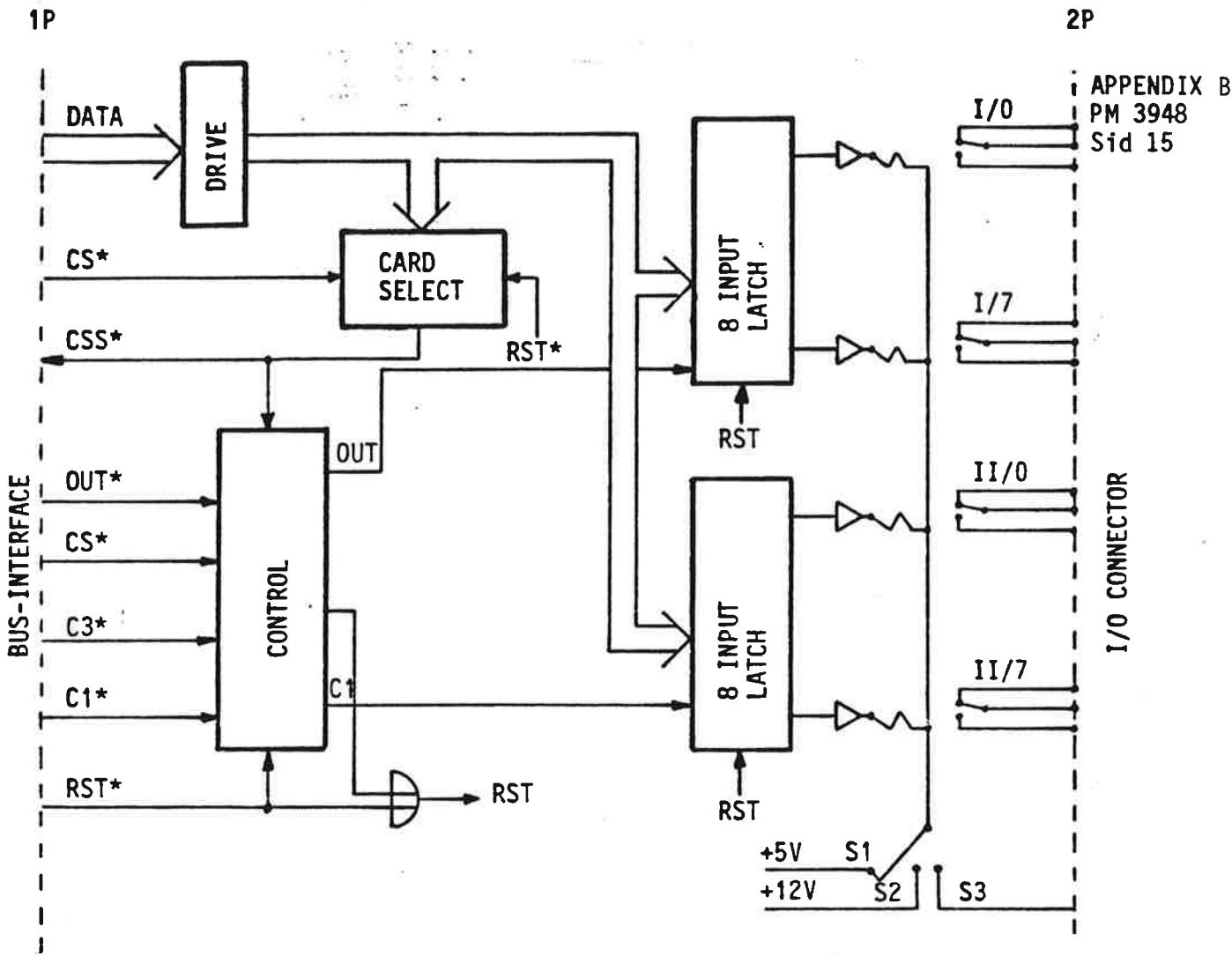
TEST-EXEMPEL

The example is written in BASIC.
It shows how to control one output at the time.
Two functions is used to set and reset bits in a variabel.
FNA(X%,Y%) resets bit number X% in variable Y%.
FNB(X%,Y%) sets bit X% in variable Y%.

Exemplet är skrivet i BASIC.
Det visar hur en utgång kan styras i taget.
Två funktioner används för noll- och ett-ställning av bitar i
variabler.
FNA(Y%,X%) Nollställer bit X% i variabel Y%.
FNB(Y%,X%) Ettställer bit X% i variabel Y%.

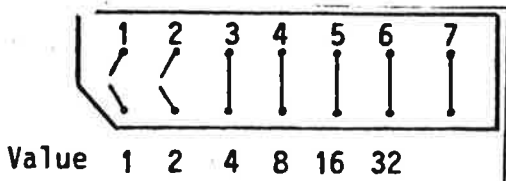
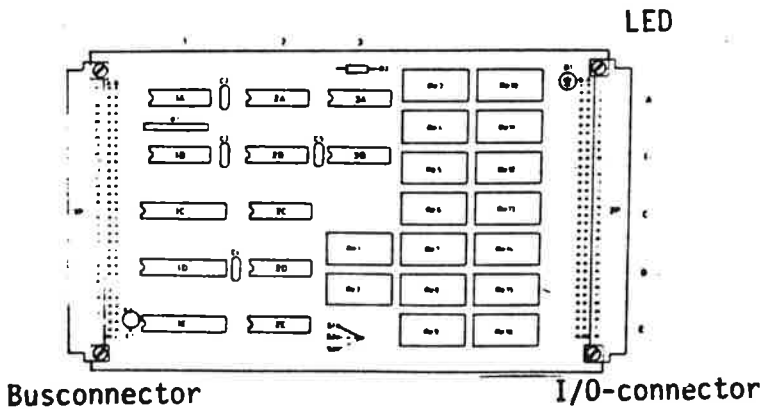
```
10 REM 4103
20 DEFFNA(X%,Y%)=Y% AND (65535%-2%*X%)
30 DEFFNB(X%,Y%)=Y% OR 2%*X%
50 OUT 1,3 : REM CARD SELECT
60 PRINT CHR$(12) : REM 4680-BASIC USE CHR$(26)
70 PRINT "WHICH RELAY (0-15)";
80 INPUT S
90 PRINT
100 IF S<0 OR S>15 THEN 70
110 PRINT "SET OR RESET (1/0)";
120 INPUT A
130 IF A<0 OR A>1 THEN 110
140 IF A=0 THEN U1=FNA(U1,S) ELSE U1=FNB(U1,S)
150 OUT 0,U1,2,SWAP$(U1)
160 GOTO 60
170 END
```

=====



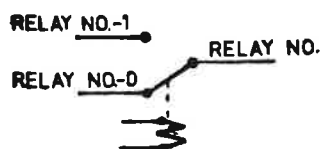
BLOCK DIAGRAM 4103

Component side Circuit board side



Cut jumpers 1 and 2 - gives address 1+2=3

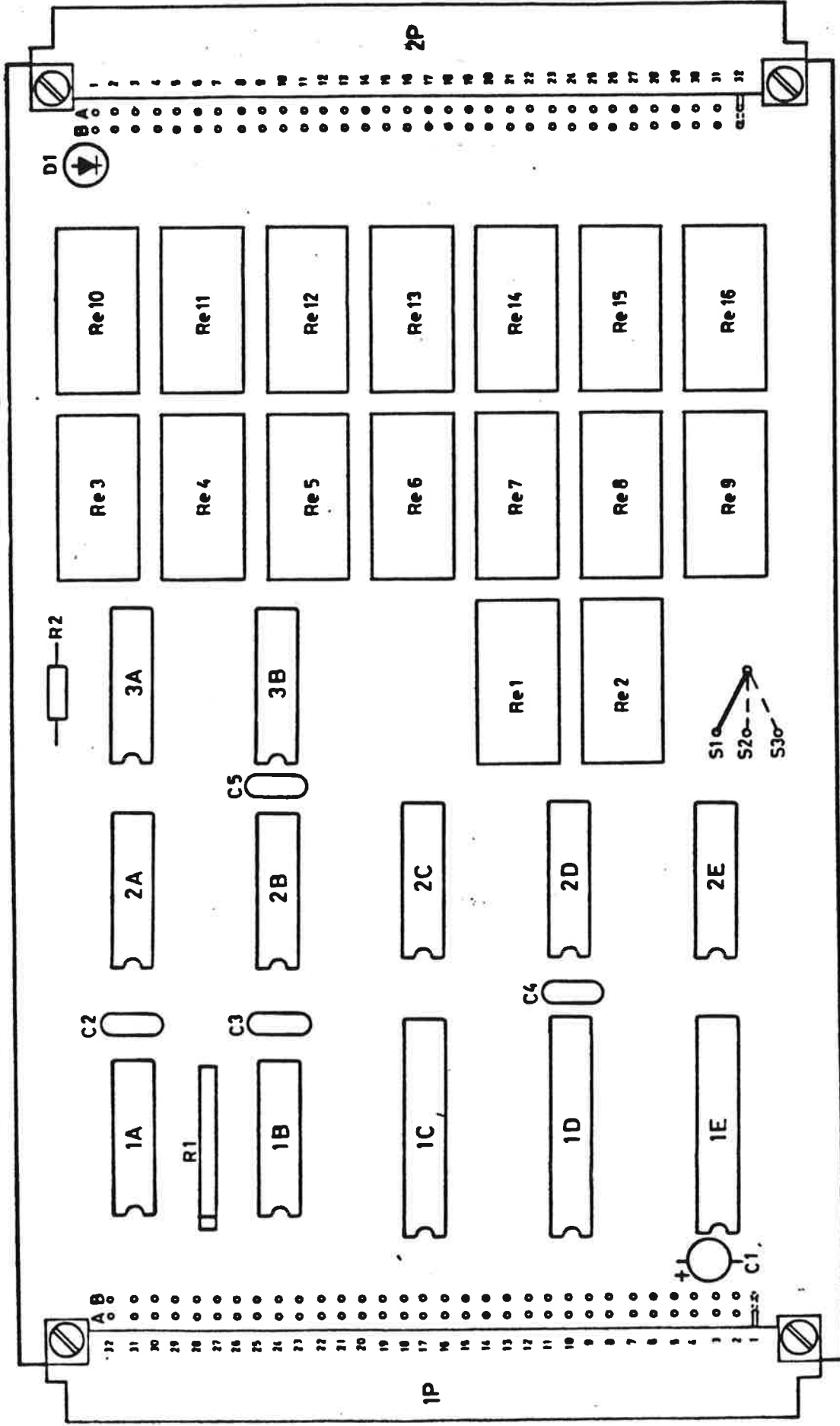
Position of code plug: 2A



Component side	Pin	Circuit board side
GND	1	GND
RELAY 0-1	2	RELAY 0 GROUP 1
RELAY 1-1	3	RELAY 1
RELAY 2-1	4	RELAY 2
RELAY 3-1	5	RELAY 3
RELAY 4-1	6	RELAY 4
RELAY 5-1	7	RELAY 5
RELAY 6-1	8	RELAY 6
RELAY 7-1	9	RELAY 7
RELAY 8-1	10	RELAY 8 GROUP 2
RELAY 9-1	11	RELAY 9
RELAY 10-1	12	RELAY 10
RELAY 11-1	13	RELAY 11
RELAY 12-1	14	RELAY 12
RELAY 13-1	15	RELAY 13
RELAY 14-1	16	RELAY 14
RELAY 15-1	17	RELAY 15
RELAY 1-0	18	RELAY 0-0
RELAY 3-0	19	RELAY 2-0
RELAY 5-0	20	RELAY 4-0
RELAY 7-0	21	RELAY 6-0
RELAY 9-0	22	RELAY 8-0
RELAY 11-0	23	RELAY 10-0
RELAY 13-0	24	RELAY 12-0
RELAY 15-0	25	RELAY 14-0
EXTERNAL SUPPLY VOLTAGE	26	EXTERNAL SUPPLY VOLTAGE
	27	
	28	
	29	
	30	
	31	
	32	

I/O-CONNECTOR

1 2 3



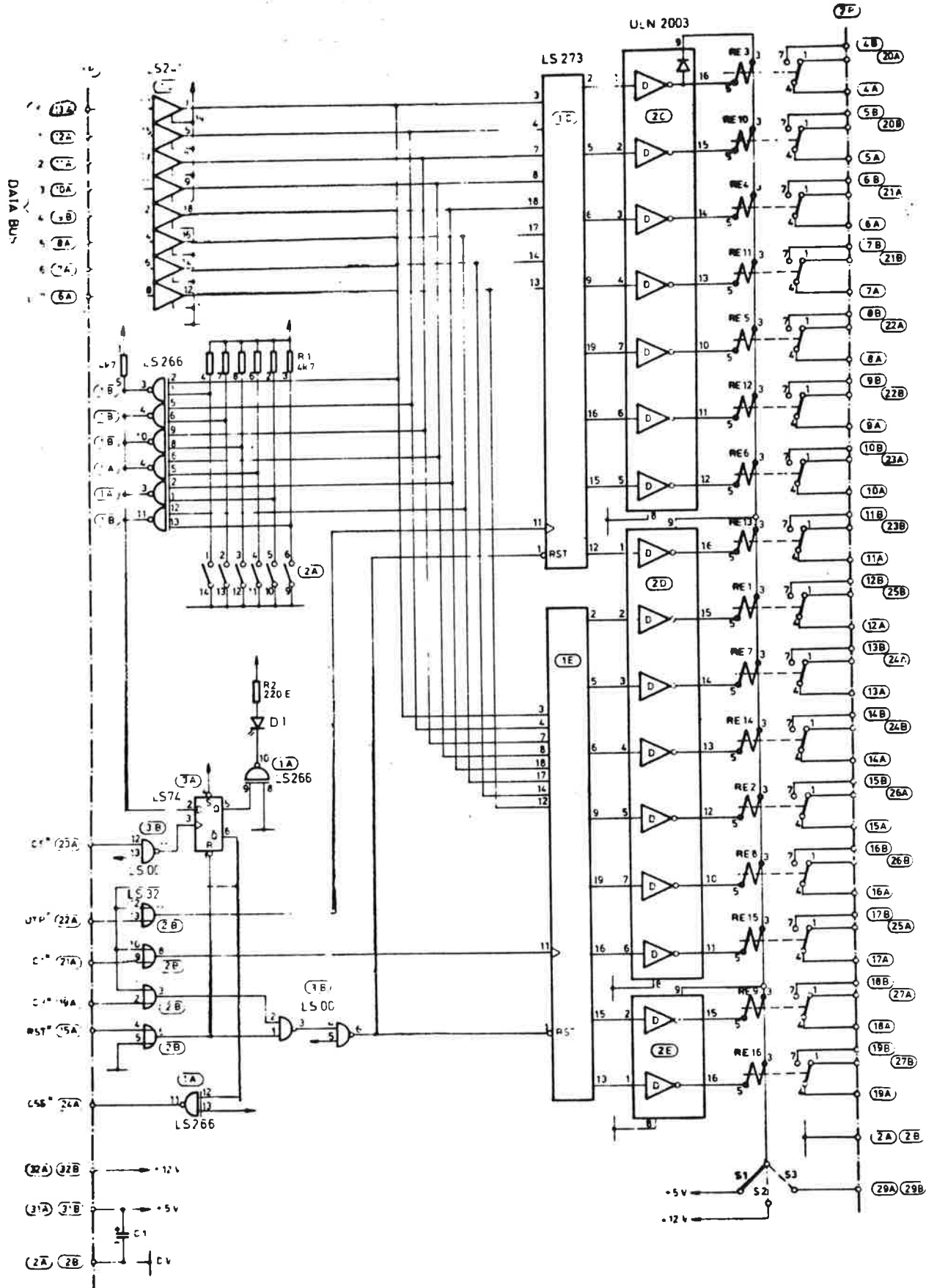
Cover strapping options S2, S3 during wave soldering

APPENDIX B
PM 3948
Sid 16

DATA INDUSTRIER AB		TÄBY SWEDEN	
A	REF	MIT	DATUM
	010409		

82-4103-00

16 RELAY OUTPUT



INDUSTRIES AG
16 URE

4103

4680 103

B.3 BETECKNINGAR FÖR GIVARE I PROGRAM

Tubnr är en variabel som antar värdena 1, 2 och 3.

Givare	Namn i program	Tub nr, kan anta
S51	S (tubnr, 51)	1, 2
S52	S (tubnr, 52)	1, 2
S53	S (tubnr, 53)	3
S54	S (tubnr, 54)	3
S55	S (tubnr, 55)	3
S56	S (tubnr, 56)	3
S57	S (tubnr, 57)	samt1
S58	S (tubnr, 58)	samt1
S60	S (tubnr, 60)	samt1
S61	S (tubnr, 61)	samt1
S62	S (tubnr, 62)	samt1
S63	S (tubnr, 63)	samt1
S65	S (tubnr, 65)	samt1
S70	S (tubnr, 70)	samt1
S71	S (tubnr, 71)	samt1
S72	S (tubnr, 72)	samt1
S73	S (tubnr, 73)	samt1
S74	S (tubnr, 74)	1, 2
S75	S (tubnr, 75)	1, 2
S80	S (tubnr, 80)	samt1
S81	S (tubnr, 81)	samt1
S82	S (tubnr, 82)	samt1
S83	S (tubnr, 83)	samt1
S84	S (tubnr, 84)	3
S85	S (tubnr, 85)	3
S86	S (tubnr, 86)	1, 2
S87	S (tubnr, 87)	1, 2
S88	S (tubnr, 88)	1, 2
S90	S (tubnr, 90)	samt1

Givare	Namn i program	Tub nr, kan anta
S91	S (tubnr, 91)	samt1
S92	S (tubnr, 92)	samt1
S93	S (tubnr, 93)	samt1
S94	S (tubnr, 94)	samt1
S95	S (tubnr, 95)	samt1
S96	S (tubnr, 96)	samt1
S97	S (tubnr, 97)	samt1

* Givarvektor *

S98	SG(98)
S99	SG(99)
S100	SG(100)
S101	SG(101)
S102	SG(102)
S103	SG(103)
S104	SG(104)
S105	SG(105)
S106	SG(106)
S107	SG(107)
S108	SG(108)
S109	SG(109)
S111	SG(111)
S112	SG(112)
S113	SG(113)
S114	SG(114)
S115	SG(115)
S116	SG(116)
S117	SG(117)
S118	SG(118)
S119	SG(119)

B.4 BILDSKÄRM

Övre delen av bildskärmen visar ventilstatus och givarstatus. En fylld ruta innebär att ventilen/givaren är gemensam för flera tuber. Den etta/nolla som står ovanför rutan gäller då även för den tub som har en fylld ruta på sin rad.

De givare som anger saltankens status har fått förkortade beteckningar. Givare S(101) anges med endast en etta, S(102) med en tvåa osv. Även de givare som betecknas som "övriga givare" har i en del fall förlorat sin första siffra. Förkortningarna är givetvis gjorda för att spara utrymme.

Den lilla fyrkant som finns ovanför torpedtuberna symboliserar säkerhetsledningen. När en torpedtub är laddad skrivs texten ovanför tuben med inverterad text. När tub X är laddad med en torped i förliga läget skrivs "X" med inverterad text. En torped i aktra läget medför att "TUB" inverteras.

Varningar, felutskrifter samt anvisningar vid kommandogivning dyker upp på den blanka raden längst ned på skärmen.

På nästa sida följer en skiss som visar hur bilden på datorskärmen ser ut efter initieringen.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	
VENTIL	STATUS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	ÖS	
1	3	5	7	9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
2	0	0	1	2	3	4	5	6	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	
7	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
11	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
12	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
13	TUB BTT		TUB TVÅ		TUB TAE		TUB TAE		TUB TAE		TUB TAE		TUB TAE		TUB TAE		TUB TAE		TUB TAE		TUB TAE		TUB TAE	
14	VENTSTRYLKA		VENTSTRYLKA		VENTSTRYLKA		VENTSTRYLKA		VENTSTRYLKA		VENTSTRYLKA		VENTSTRYLKA		VENTSTRYLKA		VENTSTRYLKA		VENTSTRYLKA		VENTSTRYLKA		VENTSTRYLKA	
15	LADDN/URT TORPED-T		MODULIGA KOMMANDON:		ÖNS TUBBLOK-L		SUDDA MÄSTA RAN-M		FRAN/BAE SÄKL-F		VENT/61 V ÅDRING-A		FRAN/BAE SÄKL-F		VENT/61 V ÅDRING-A		FRAN/BAE SÄKL-F		VENT/61 V ÅDRING-A		FRAN/BAE SÄKL-F		VENT/61 V ÅDRING-A	
16	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
17	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
18	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
19	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
20	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
21	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
22	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
23	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Tot	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23

Mottagare

C. PROGRAMKÖRNING

C.1 OPERATÖRSINSTRUKTIONER

En programkörning inleds med:

- a) Slå till dator och skivenhet
- b) Sätt skivan i undre "diskdriven"
- c) skriv "RUN UPPSTART" RETURN

Datorn ställer därefter ett antal frågor som skall besvaras enligt givna instruktioner. Observera att svaren måste ges med versaler.

När alla frågorna besvarats ritas den grafiska presentationen upp på bildskärmen. Den övre delen visar ventilstatus och givarstatus. I mitten är de tre torpedtuberna uppritade. Där visas grafiskt luckornas lägen och vattennivån i tuberna. Den lilla fyrkanten ovanpå tuberna symboliserar säkerhetsledningen. När en torpedtub är laddad skrivs texten ovanför tuben med mörk text mot ljus bakgrund, s k inverterad text. Trycket i tuberna är p g a det begränsade programutrymmet uppdelat i tre diskreta nivåer. Aktuellt tryck visas under tuberna. Längst ner på skärmen syns de grundkommandon som är möjliga under exekveringen.

Huvudprogrammet anropas så snart skärmen har fyllts. Programmet "UPPSTART" försvinner då ur datorn och huvudprogrammet "HUVUD" läses in.

Datorn är nu redo att ta emot kommandon från tangentbordet och eventuella ventilsignaler från kortracken. De kommandon som matas in via tangentbordet måste ges stegvis enligt de instruktioner som dyker upp på nedersta raden. Detta beror på att tangentbordsbufferten endast rymmer ett tecken och det faktum att programmet aldrig får stanna upp.

I grundläget finns det sex olika kommandon att välja mellan. Det kommando som väljs skrivs med inverterad text på skärmen. De valmöjligheter som finns i nästa steg skrivs samtidigt ut på nedersta raden. Texten förblir inverterad till dess operatören har lämnat all information som behövs för att datorn skall kunna utföra instruktionen.

Nedan följer förklaringar till de kommandon som kan ges från tangentbordet.

Laddning och urtagning av torped = T.

Det finns ett antal olika torpedtyper. När en tub är laddad skrivs texten ovanför tuben med inverterad text. Om två torpeder laddas i en tub inverteras första halvan av texten ovanför tuben när aktra torpeden är laddad och sista halvan då förliga torpeden laddas. Tänk på att vissa torpeder behöver aktiveras med hjälp av tryckluft. Detta sker med ventil nr 15.

Öppning och stängning av tublock = L

Säkerhetsledningen måste vara i förliga läget för att luckan skall kunna öppnas. Det går inte att öppna luckan när tuben är trycksatt eller när det finns vatten i tuben.

Säkerhetsledningen flyttas framåt eller bakåt = F

Säkerhetsledningen spärrar luckan i ena änden av torpedtuben. Mynningsluckan kan t ex ej öppnas om säkerhetsledningen är i det förliga läget.

Sudda nedersta raden = U

Används för att bli av med felmeddelanden och varningar. Om texten återkommer efter raderingen krävs det någon åtgärd från operatörens sida för att eliminera orsaken till texten.

Ändra manuella ventiler eller vissa givare = Ä

Ventilerna 20, 21 och 22 ändras manuellt ombord på ubåten. Dessa tre salvtanksventiler kan därför inte styras från tubdatorn. Vid tangentbordsstyrning (= S) styr man endast de ventiler som vid drift styrs av tubdatorn.

De givare som vi kan ändra på direkt har följande uppgifter:

S100	Anger att tryckskrovsventilen V9 är klargjord. (V9/V18 går ej att öppna om de ej är klargjorda.)
S108	Tryckskrovsventilen V18 är klargjord
S114	Indikering automatsäkring Tub 1
S115	- " - Tub 2
S116	- " - Tub 3
S117	- " - Övriga säkringar

Tryckskrovsventilerna klargörs helt enkelt genom att man ändrar respektive givare. Det är lika lätt att förstöra en säkring.

Mottagare

Tangentbordsstyrning = S

I uppstartningsprogrammet väljer operatören den metod som önskas för att styra ventilerna. När man valt att ventilsignalerna skall komma från korttracken går det inte att styra från tangentbordet. Datorn accepterar endast ett styrsätt.

Datorn får ventilsignalerna utifrån i form av 8-bitars binära tal, där varje ventil representeras av en bit. Denna representation har legat till grund även för tangentbordsstyrningen. Vid tangentbordsstyrning sker valet av ventil och önskad ändring stegvis.

Först anges önskad byte, därefter aktuell bit och till sist, om ventilen skall öppnas (1) eller stängas (0).

När datorn mottagit hela kommandot sker ett hopp till respektive subrutin där orderns lämplighet testas innan den utförs. Om ändringen är mindre lyckad dyker det upp en varningstext innan ändringen utförs. Uppstår ett allvarligt fel när ordern utförs ignoreras ordern och en förklarande text skrivs ut. Detta upprepas till dess ordern återtas eller det som orsakat datorns missnöje har undanröjts. Är felet så allvarligt att ubåten vattenfylls fastnar programmet i subrutinen "VATTEN I UBÅT". Så fort operatören trycker ner en tangent återupptas exekveringen.

Hur de bytes som datorn arbetar med ser ut framgår av Appendix B. Här nedan följer en mer användbar version. Det första av de ensiffriga talen som står i tabellen är aktuell byte. Nästa siffra anger den bit som är av intresse.

	<u>Tub nr 1</u>	<u>Tub nr 2</u>	<u>Tub nr 3</u>
V1	1 0	3 0	5 0
V3	1 1	3 1	5 1
V5	1 2	3 2	5 2
V7	1 3	3 3	5 3
V9	7 6		
V10Ö	1 4	3 4	5 4
V10S	1 5	3 5	5 5
V11	1 6	3 6	5 6
V12Ö			5 7
V12S			6 0
V13			6 1
V14Ö	1 7	3 7	
V14S	2 0	4 0	
V15	2 1	4 1	
V16Ö			6 2
V16S			7 0
V17Ö			7 1
V17S			7 2

Mottagare

Här följer till sist några tips som kan underlätta operatörens arbete.

Kommandot "SAVE HUVUD" gör att programmet "HUVUD" sparas på flexskivan. Programmet sparas då i s k BAC-form. "KILL MO1:HUVUD.BAC" raderar samma program om det ligger i undre "Diskdriven". Annars bytes "MO1:" mot "MOØ:"

Utskrift på den printer som är tillgänglig på Kockums sker med kommandot: "LIST PR:00020C00.000".

Avfyring av torpederna sker då torpedlåsen lyfts.

Uppstår problem vid listning eller utskrift via printer - skriv kommandot "INTEGER".