

STYR
OCH
MÄT
MED ABC 80

av
Åke Westh

Studentlitteratur

Till Lennart
med hälsning
från Jörg.
Elin.

STYR och MÄT med ABC80

av

Åke Westh

Studentlitteratur

De i boken förekommande konstruktionerna
får ej användas i kommersiellt syfte.

© Åke Westh 1980
Printed in Sweden
Studentlitteratur
Lund 1980
ISBN 91-44-17631-7

Innehåll

Förord	5
V:24 snittet	7
Programmering av V:24	9
Tillämpningskopplingar för V:24-snittet	14
Styrning av display	14
Inläsning från display	15
Styrning av nio switch-funktioner	17
Processkontroll	23
Logik monitor	29
ABC-bussen	35
Expansionsmöjligheter med ABC80	39
ABC80 – minne och minnesexpansion	45
Minnesstacken	45
Minnesexpansion	48
Adressering av I/O-kort	52
Anslutning av snabbskrivare	54
Initiering av printer	54
Åtta reläutgångar	58
Inkoppling	58
I/O-kommandon för 4007	60
Laborationer med reläkort	60
Analog-digitalomvandlare AD	63
Inkoppling av AD	64
I/O-kommandon för 4082	66
Laborationer på AD	67
DAC	69
I/O-kommando 4083	72
ABC80 som funktionsgenerator	82
System för komponentprovning	84
Automatisk mätvärdesinsamling	86
IEC-bussen	91
Bussens struktur	94
Adressval	96
Programmering av IEC-bussen	99
Inkoppling av IEC-utrustningen	101

Mätning med IEC-buss	103
IEC-bussmätning av frekvensgång	107
Bilaga 1: Kopplingsschema 12" display (ABC80)	117
Bilaga 2: Kopplingsschema datorenhet	119
Bilaga 3: Kopplingsschema ABC 80 kassetminne	121
Referenslitteratur	123
Sakregister	124

Förord

Med föreliggande bok vill jag rikta uppmärksamheten på hur datorn kan användas som systemkomponent. Boken vänder sig i första hand till gymnasieskolans tekniskt inriktade linjer, motsvarande arbetsmarknadsutbildningar samt till industrin. För att kunna tillgodogöra sig bokens innehåll fordras kunskaper i BASIC-programmering, digitalteknik och grundläggande data-lära.

Bokens innehåll och uppläggning speglar en del av den undervisning som har bedrivits på den högre specialkursen i kommunikationselektronik vid Vasagymnasiet i Arboga. Jag anser att verklighetsanknuten undervisning är viktig. Eleverna har därför projektarbetat. Genom denna arbetsform tvingas de till samarbete och eget kunskapssökande. Arbetet kräver också laborerande och tillämpningsövningar. Eftersom motsvarande arbeten i arbetslivet kräver dokumentation och redovisning har stor vikt lagts vid dessa moment.

Första delen av boken behandlar V:24-snittet. I detta avsnitt beskrivs ett antal styr- och mättillämpningar. Dessa tillämpningar är resultat av elevernas projektarbeten där uppgiften har varit att utveckla såväl hård- som mjukvara. De framtagna konstruktionerna i V:24-avsnittet finns inte att tillgå på den öppna marknaden.

Min förhoppning är i första hand att såväl lärare som elever ska få glädje och nytta av boken. Dagens dataundervisning får inte stanna vid enbart BASIC-programmering utan bör också omfatta systemtillämpningar. Självfallet hoppas jag även att boken ska bli till stor nytta för dagens industri. Min tredje förhoppning är att boken ger många impulser till att utöka resurserna kring ABC80 så att dess användningsområden bättre kan utnyttjas i framtiden.

Åke Westh



V:24-snittet



Bild 1.

V:24 eller "modemstandard" är till väsentliga delar jämförbar med den amerikanska standarden RS 232 C. V:24 är ett spänningsmatat serieinterface. Det medger inkoppling av utrustningar på ett avstånd från ABC80 av max 15 meter. Den vanligaste användningen av V:24-snittet är anslutning av printar, digitaliseringsbord, plotter eller ljuspenna. En annan viktig funktion får V:24 snittet då ABC80 skall användas som terminal för körning mot en annan dator. Detta möjliggörs genom att till V:24-snittet ansluta ett modem. I många sammanhang önskar man använda datorn för att ge respektive ta emot signaler, digitala eller analoga, från styr-, kontroll- och övervakningsutrustningar av skilda slag. Utrustningar som på detta sätt kan anslutas till ABC80 utgör I/O-enheter. I står för input och O för output. ABC80 har två möjligheter till anslutning av I/O-enheter för styr- och mätändamål, V:24-snittet och ABC-bussen. Här ger ABC-bussen de största möjligheterna. Men det finns all anledning att uppmärksamma V:24-snittets möjligheter. Utvecklingen, framför allt på fiberoptikens område, pekar mot att den seriella bussen kommer att bli förhärskande. Detta skulle leda till en intresseförskjutning från de parallella bussarna. ABC-bussen liksom IEC-bussen är exempel på parallella bussar.

Programmering av V:24

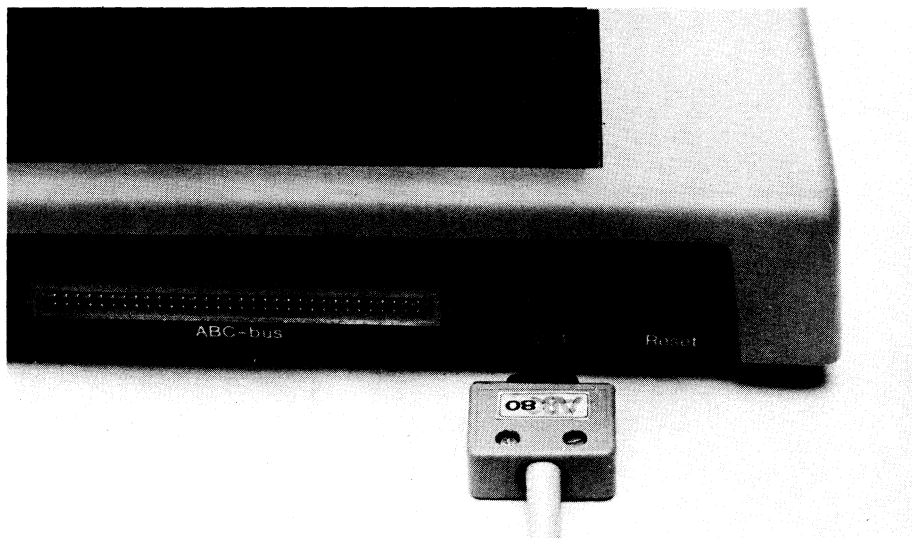


Bild 2. ABC80. Datorenhetens kontaktlist med det seriella spänningsnittets 9-poliga kontakt.

V:24-kontakten på ABC80:s baksida styrs från en PIO-krets, som har två portar A och B. V:24-snittet styrs av B-porten, som omfattar åtta bitar. Dessa är fördelade på följande sätt:

Bit 0 till 4	V:24
Bit 5 till 7	Kassetminne

Av schemat på bild 3 framgår att bit 3 och 4 är linjeutgångar som via linjedrivkretsarna LS1 och LS2 är kopplade till stift 4 och 2 på V:24-kontakten. Drivkretsarna ger en utspänning av ca 7 volt och tål en belastning av 10 mA. Observera att drivkretsarna har inverterad utgång. På samma sätt matas bitarna 0, 1 och 2 av linjemottagarna LM1 till LM3. Den översta drivkretsen har jordad ingång, vilket betyder en konstant utspänning på stift 1. Om bit 3 och 4 ligger ”höga” dvs ett-ställda, kommer motsvarande stift på V:24-kontakten att ligga ”låga” dvs nollställda. Av samma skäl kommer bit 0, 1 och 2 att vara ett-ställda om ingen spänning ligger ansluten till motsvarande kontaktstift. För att kunna

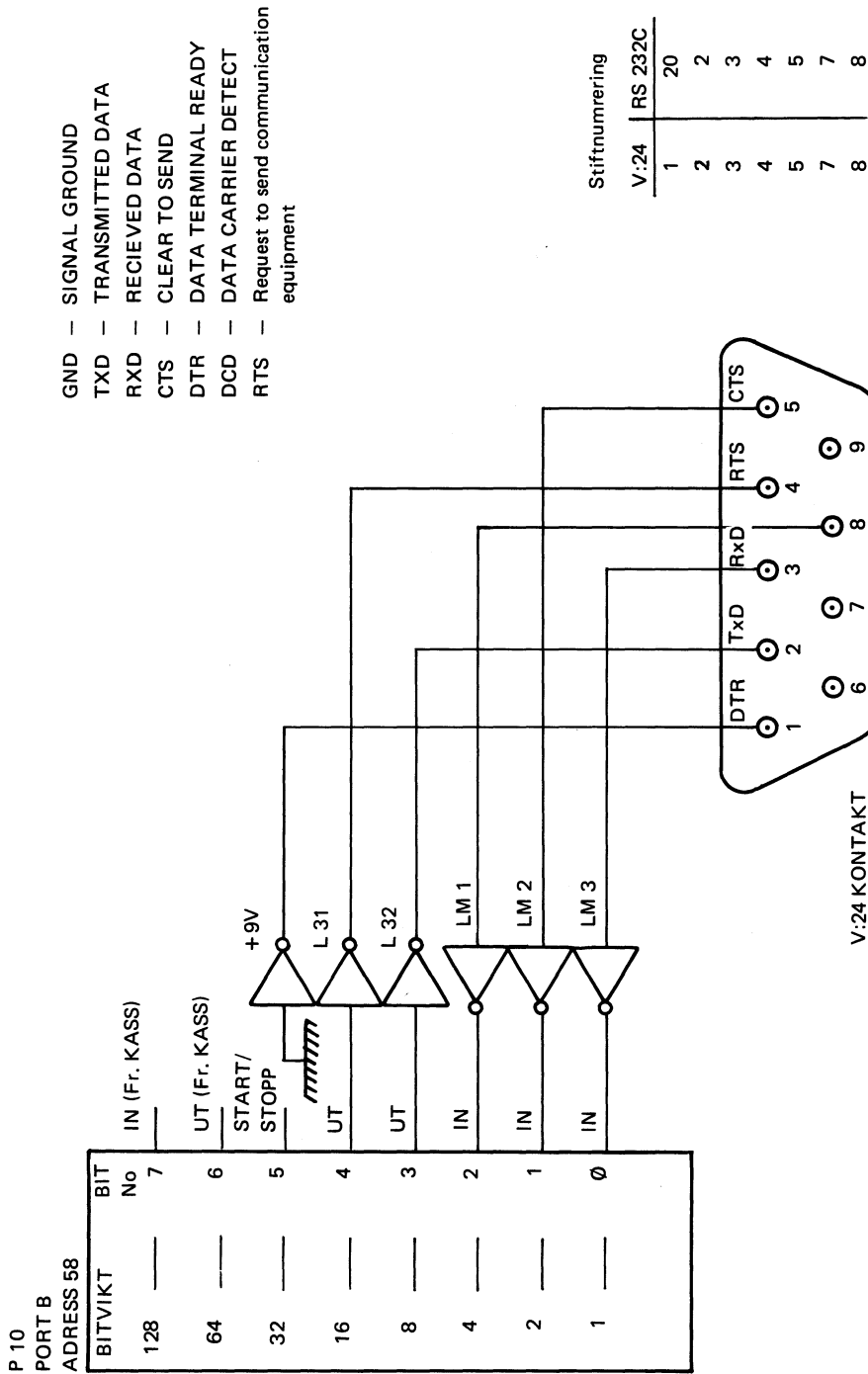


Bild 3. Schema över V:24-snittet.

ett- respektive noll-ställa de båda linjeutgångarna, som korresponderar med bit 3 och 4, används instruktionen

OUT 58,X

där X är ett värde som kan väljas mellan 0 och 255 enligt:

OUT 58,X	START- STOPP- RELÄ	BIT 4	BIT 3	V:24 STIFT 4	V:24 STIFT 2
0—7	FRÅN	0	0	1	1
8—15	FRÅN	0	1	1	0
16—23	FRÅN	1	0	0	1
24—31	FRÅN	1	1	0	0
32—39	TILL	0	0	1	1
40—47	TILL	0	1	1	0
48—55	TILL	1	0	0	1
56—63	TILL	1	1	0	0

Ex

OUT 58,15

På V:24-kontakten kommer nu stift 4 att vara ett-ställt medan stift 2 är noll-ställt.

De första bitarna 0, 1 och 2 används för att läsa in data från V:24-kontakten. Den instruktion som nu gäller är

INP (58)

Skriv

PRINT INP(58)

Datorn skriver nu ut från port B med adress 58 i decimal form. Vid läsning från porten erhålls både in- och utbitarna.

Skriv

OUT 58,0

Alla utbitarna är nu noll-ställda. Skriv därefter

PRINT INP(58)

På skärmen skrivs nu värdet 135. Detta utgör summan av bitvikterna för samtliga IN-bitar, 0, 1, 2 och 7 (se bild 3). De övriga bitarna, UT-bitarna förblir

noll-ställda. Skriv

OUT 58,8

och därefter

PRINT INP(58)

Nu blev även bit 3, med bitvikten 8, ett-ställd. Utskriften blir därför 143 som är 135+8.

För att få bort utbitarnas värde vid läsning från V:24 kan man skriva

Z=7 AND INP(58)

Z är det decimala värdet av bitvikterna för bitarna 0, 1 och 2. Skriv

Z=7 AND INP(58): PRINT Z

Bildskärmsutskriften blir nu 7 oavsett vilka data som finns på bitarna 4—7. Koppla ihop stift 8 och 1 på V:24-kontakten. Bit 2 blir nu noll-ställd genom "ettan" på stift 1. Skriv på nytt

Z=7 AND INP(58): PRINT Z

Tillståndet på bitarna 0, 1 och 2 skall nu vara

BIT	LOGISK NIVÅ	BITVIKT
0	1	1
1	1	2
2	0	4

Summan av bitvikterna blir 3 vilket också utskriften skall bli. OBS! Koppla inte ihop matningsspänningarna på stift 6 och 9 med varandra eller med jord på stift 7. Skriv in följande program:

1 PRINT 7 AND INP(58): GO TO 1

I bildskärmens vänsterkant skriver nu datorn 7. Prova nu att ett-ställa ett eller flera av stiften 3, 5 och 8 på V:24-kontakten. "Ettan" finns på stift 1. Kontrollera mot tabellen.

V:24			PIO B			
STIFT 8	STIFT 5	STIFT 3	BIT 2	BIT 1	BIT 0	7 AND INP(58)
0	0	0	1	1	1	7
0	0	1	1	1	0	6
0	1	0	1	0	1	5
0	1	1	1	0	0	4
1	0	0	0	1	1	3
1	0	1	0	1	0	2
1	1	0	0	0	1	1
1	1	1	0	0	0	0

I den resterande delen av V:24-sektionen beskrivs ett antal styr- och mättilämpningar. Dessa är resultatet av projektarbeten utförda av elever på den högre specialkursen i kommunikationselektronik vid Vasagymnasiet i Arboga. Uppgiften har varit att utveckla såväl hård- som mjukvara. Eftersom dessa konstruktioner inte finns att tillgå på den öppna marknaden är min förhoppning att boken ger impulser så att resurserna kring ABC80 utökas. Detta är angeläget, om arbetet med datorn inte skall begränsas till enbart BASIC-programmering. ABC80 i grundutförande är det enda som krävs för att kunna använda dessa konstruktioner.

Tillämpningskopplingar för V:24-snittet

Styrning av display

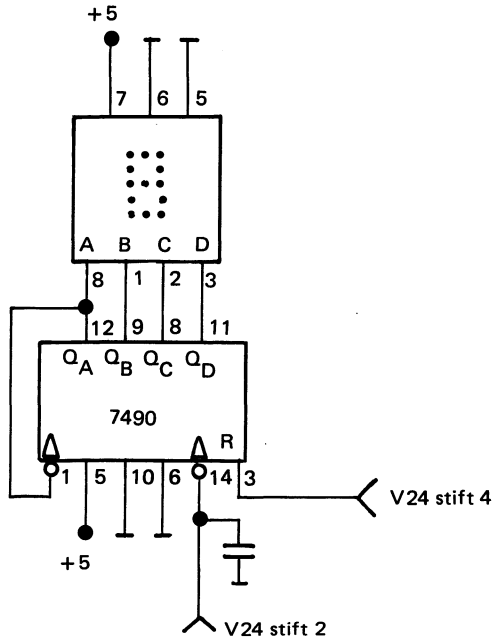


Bild 4. Schema över displaykoppling avsedd för anslutning till V:24.

Grundkopplingen, som framgår av bild 4, gör det möjligt att från ABC80 styra en information till en extern display. Kopplingen, som kan anslutas på ett antal meters avstånd från ABC80, är här begränsad till visning inom sifferområdet 0 till 9. Den siffra, som displayen visar, svarar mot nedtryckt siffertangent på ABC80. Kopplingen består av en dekadräknare (7490) och en display med inbyggd avkodare (5082—7300). Stift 14 på dekadräknaren är klockingång och matas av pulser från stift 2 på V:24-kontakten. Eftersom kopplingens strömförbrukning är liten kan denna ske från V:24-kontakten. På stift 6 finns en 12 volts spänning som belastas med 100 mA. Om en krets (7805) kopplas mellan

stift 6 och kopplingen erhålls 5 voltsmatning. Anslut kopplingen och mät upp matningsspänningen på respektive krets. Skriv därefter följande program.

```
1 REM PROGRAM FÖR STYRNING AV DISPLAY VIA V24 : KONTAKTEN
2 OUT 58,8
3 ; CHR$(12)"ANGE EN SIFFRA 0-9";
4 ONERRORGOTO 3 : GET A# : X=VAL(A#)
5 FOR I=Y TO X
6 OUT 58,24
7 OUT 58,16
8 NEXT I
9 ; CHR$(12)"RESET (J/N) ?" : GET R#
10 IF R#<>"J" AND R#<>"N" THEN Y=1 : GOTO 3
11 OUT 58,8 : Y=0 : GOTO 3
```

På rad 2 noll-ställs displayen.

3 frågar efter en sifvertangentsnedtryckning,

4—8 utgör en pulsgenerator, som avger det antal pulser, vilka svarar mot det värde variabeln X har tilldelats,

11 datorn ger en reset-puls till dekadräknaren.

Inläsning från display

Med den förra kopplingen tjänstgjorde datorn som sändare av en information, vilken mottogs av en display. Motsatsen skall ske med nästa koppling. Datorn skall alltså vara mottagare av en information som kopplingen på bild 5 ska lämna. Med kopplingens tangent S kan ett tvåsiffrigt värde ställas in på displayen. Enheten kan sedan anropas av ABC80 och värdet läses in för att lagras i en variabel eller skrivs ut på skärmen. Tekniken visar principen för ett datainsamlingssystem där ett antal enheter av denna typ, via en multiplexer, kan anropas från datorn. För att öka betjäningsbarheten ersätts tangenten med ett numeriskt keyboard, som via en encoder-krets underlättar värdeinmatningen.

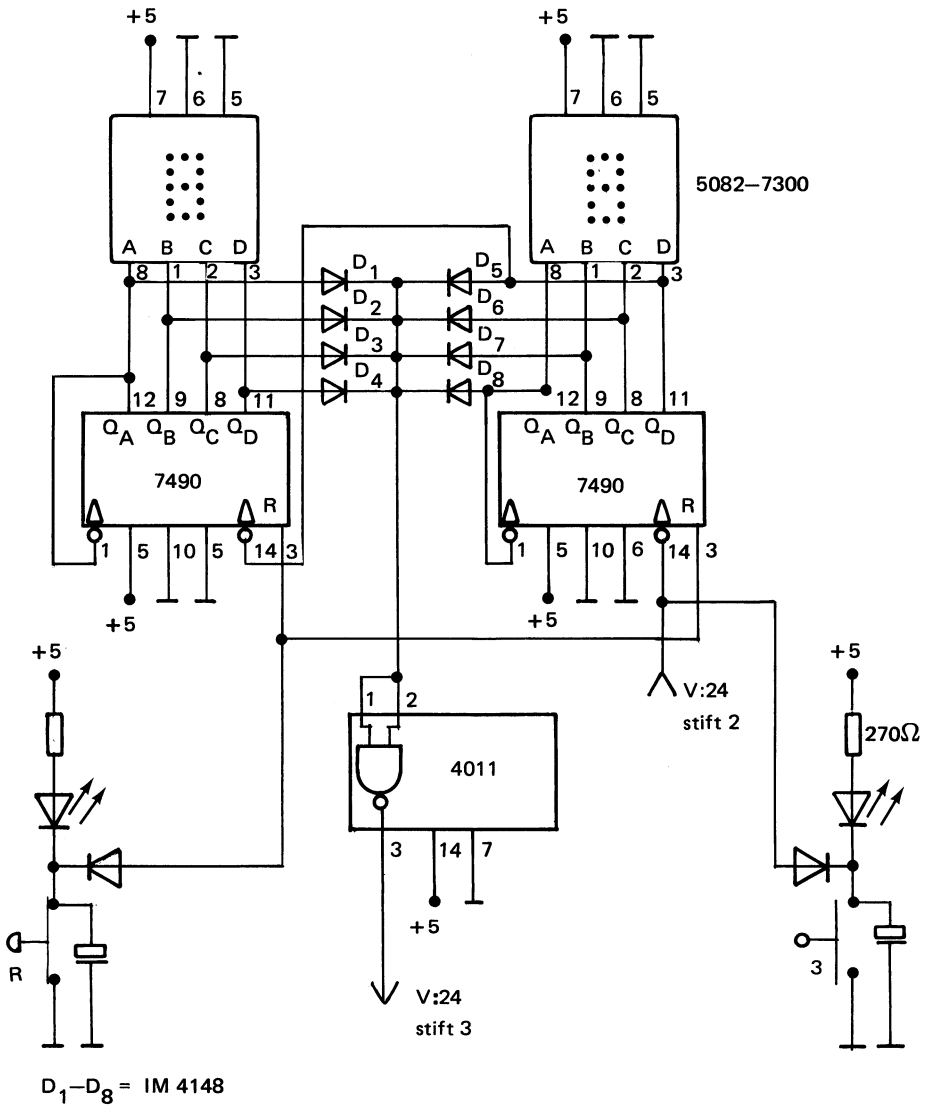


Bild 5. Schema över displaykoppling. Med tangenten S kan ett värde tryckas fram på displayvärdet in till datorn. Då ABC80 anropar enheten läses displayvärdet in till datorn.

Om displayen ska visa t ex värdet 15 stegas 15 pulser in till klockingången med tangenten S. När RETURN-tangenten på ABC80 trycks ner skickas ett pulståg från datorn via stift 2 på V:24-kontakten till samma klockingång vilken tangenten S är ansluten till. Efter 85 pulser, till de redan inmatade 15, är

tillståndet på dekadräknarnas utgångar noll-ställt. Enligt programrad 8 i programmet nedan sker utskrift av det komplementära talet till 85, dvs 15.

```
1 REM PROGRAM FÖR INLÄSNING AV TAL FRÅN DISPLAYEN
2 ; CHR$(12)"TRYCK FRAM ETT TAL PÅ DISPLAYEN 0-99 "
3 ; ; "TRYCK PÅ RETURN NÄR DU ÄR KLAR "; : GET G$
4 FOR I=1 TO 100
5 OUT 58,24
6 OUT 58,16
7 N=7 AND INP(58)
8 IF N=6 THEN ; CHR$(12)"TALET ÄR "100-I
9 NEXT I
10 ; ; "TRYCK PÅ RETURN"; : GET G$ : GOTO 2
```

Datorn fortsätter nu enligt programmet i sin loop, att skicka det antal pulser som återstår för att loopen skall genomlöpas 100 gånger. I det valda exemplet betyder det att ytterligare 15 pulser skickas till klockingången på dekadräknaren. Displayerna visar därför, även efter inläsningen, det värde som de från början var ställda att visa.

Styrning av nio switch-funktioner

Tillämpningen, som här skall beskrivas, kan styra nio individuella switch-funktioner (till-från). Till de nio utgångarna, som är av open-collector-typ, kan kopplas mindre reläer, optokopplare, lampor och dyligt.

Kopplingen, som framgår av bild 6, styrs från V:24-snittet och har nio buffrade utgångar. Parallellt med utgångarna ligger nio lysdioder för att markera vald utgång. Programmet, som driver kopplingen, medger att en eller flera valfria utgångar samtidigt aktiveras. Utgångarnas till- och frånslag kan dessutom fördröjas med valfri tid.

För att ett-ställa en utgång, t ex utgång fem, fordras att ABC80 alstrar fem pulser, som via V:24-kontaktens stift 2 matar dekadräknarens (7490) klockingång. Så snart pulståget har alstrats följer från ABC80 en kort puls, som matas till enable-ingången på latches (7475). Utgången (5), dvs stift 6 på dekoderkretsen (7442) går nu "låg", vilket gör att flip-flop-kretsen (5)(7476) kantraras. Q-utgången på flip-flop-kretsen (5) noll-ställs och lysdioden (5) tänds. (I serie med lysdioden ligger en strömbegränsningsresistor). Samtidigt kommer Q-utgången på samma krets att ett-ställas. Q-utgångarna är kopplade till någon av enhetens utgångar. Interfacekretsen har två TTL-kompatibla-ingångar och en drivtransistor med open-collector som utgång.

ABC80 fortsätter att skicka ut pulser till klockingången på dekadräknaren. Efter varje pulståg kontrollerar datorn om utgångarna på dekadräknaren är noll-ställda. Om så är kommer bit 2 på PIO:n att ett-ställas genom att stift 8 på V:24-kontakten noll-ställs. Därmed är också dekadräknaren noll-ställd. ABC80 alstrar en ny puls till latches så att även den noll-ställs. Alla utgångar,

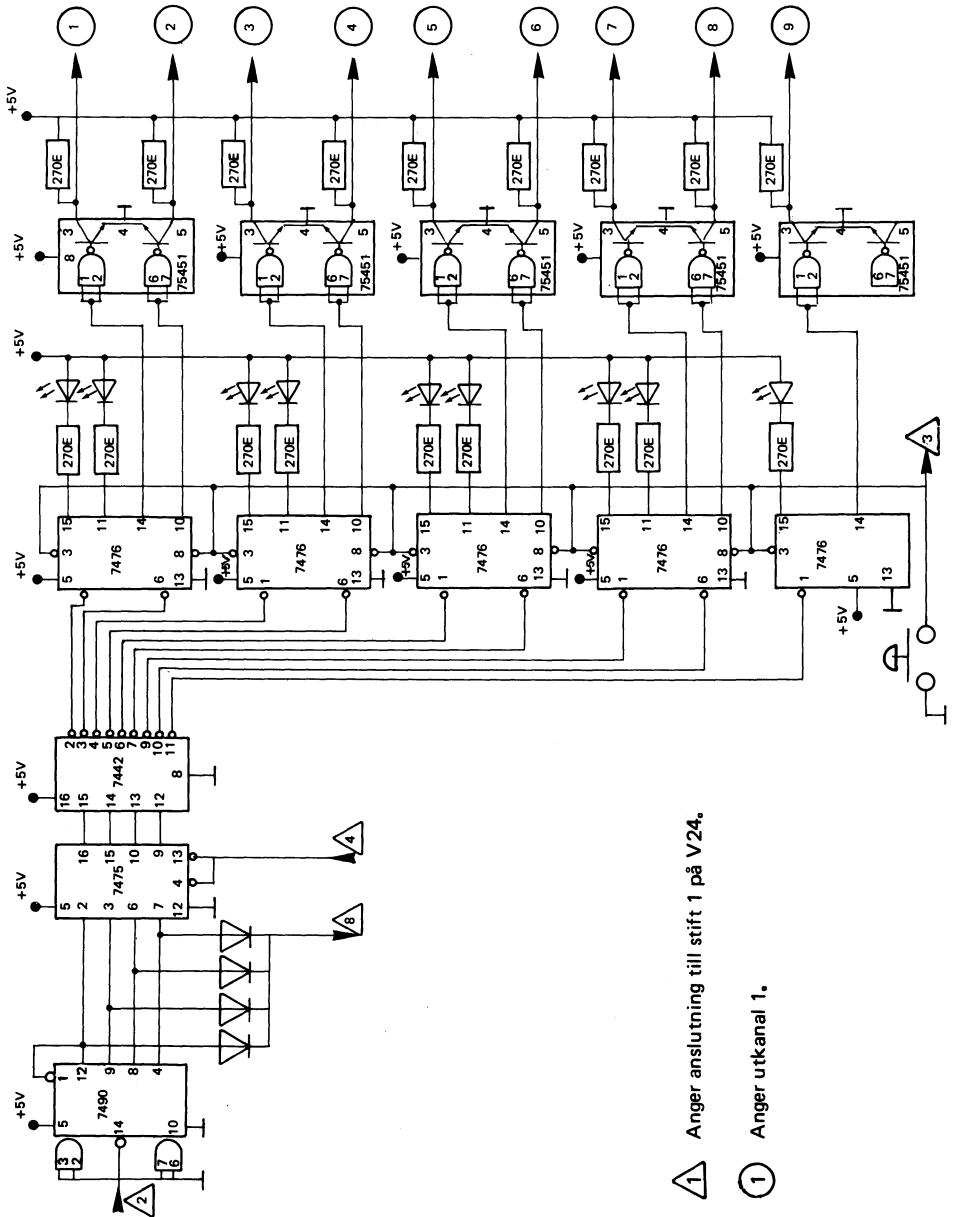


Bild 6. Kopplingen, som styrs från V:24, har nio individuellt styrbara utgångar.

utom utgång 0 på kretsen 7442, ett-ställs. Kopplingen, som drar ca 200 mA fordrar yttre spänningskälla.

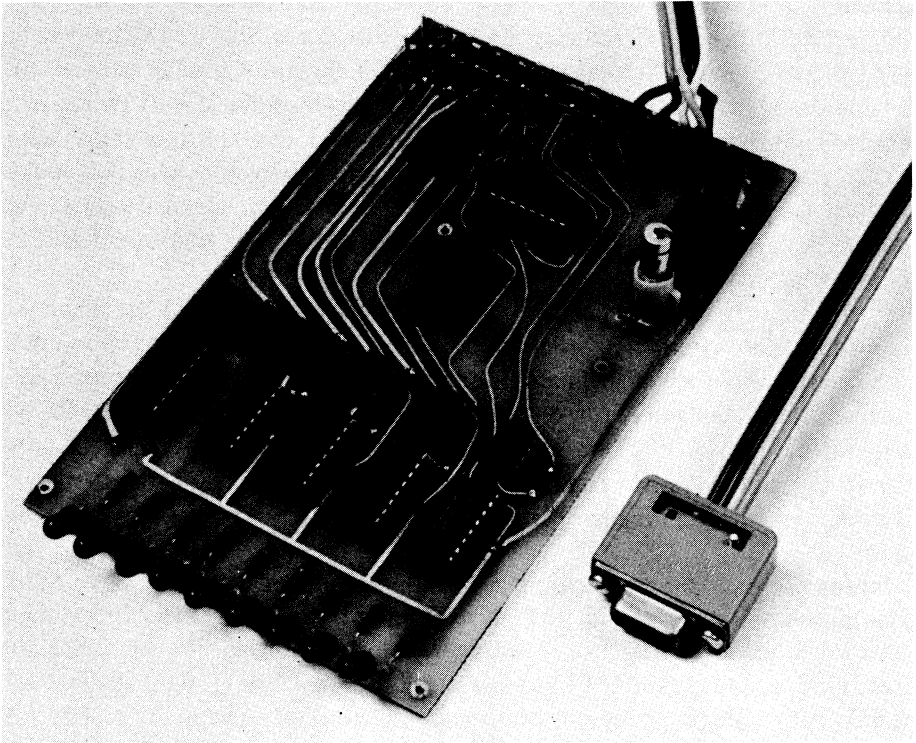


Bild 7. Exempel på hur styrkortet kan byggas upp.

Skriv in följande program som medger manuell styrning av kopplingen. Lagra för säkerhets skull programmet på kassett/flexskriver. I programmet är utgångarna benämnda 1 till 9.

```
10 REM PROGRAM FÖR MANUELL STYRNING AV 9 SWITCHFUNKTIONER VIA V24 : KONTAKTEN
20 GOTO 120
30 ; CHR$(12)"ANGE EN SIFFRA 1-9";
40 ONERRORGOTO 30 : GET A$ : IF A$="0" THEN 30 : X=VAL(A$)
50 FOR I=1 TO X
60 OUT 58,16
70 OUT 58,24
80 NEXT I
110 OUT 58,8
120 OUT 58,16
125 Z=7 AND INP(58)
130 IF Z=6 THEN OUT 58,0 : GOTO 30
140 OUT 58,24 : GOTO 120
```

På programrad 40 läser datorn av tangentnedtryckningen från någon av siffertangenterna 1—9 och tilldelar variabeln X motsvarande värde. Loopen på raderna 50—80 genomlöps X gånger och alstrar därigenom X pulser. På rad 60 matas dessa via stift 2 på V:24-kontakten till dekadräknarens klockingång. För att verkligen få ett pulståg noll-ställs stift 2 av rad 70. Då loopen är genomlöst X gånger kommer rad 110 att ett-ställa stift 4 som är kopplad till enableingången. Rad 120 ett-ställer stift 2 och noll-ställer stift 4. På rad 130 kontrollerar datorn om dekadräknaren är noll-ställd, dvs om stift 8 är noll. Om så är fallet ett-ställs stift 4, vilket resulterar i att latchens utgångar noll-ställs. Datorn går nu tillbaka till rad 30 och frågar efter en ny instruktion. Är dekadräknaren inte noll-ställd går datorn till rad 140 som noll-ställer stift 2. Därefter går datorn tillbaka till rad 120.

Program och koppling är nu klara för kontrollkörning. Börja med att kontrollera att styrenheten är kopplad till V:24-kontakten och att styrenheten har drivspänning +5 volt. Innan programexekveringen börjar kan tryckknappen på kortet användas för resetning av kopplingen. Skriv därefter RUN. Programmet börjar med att fråga vilken utgång som skall påverkas. Genom att trycka ner en siffertangent, t ex 3, kommer utgång 3 att ett-ställas. Ytterligare en tryckning på samma tangent resulterar i att samma utgång noll-ställs.

Nästa program är en utveckling av grundprogrammet. Detta gör det möjligt att automatiskt styra till- och frånslag av en eller flera utgångar samt bestämma tidsfördröjning för respektive utgångar. I programlistans början finns en identifieringsinformation. Programmet möjliggör lagring, på kassett eller flexskiva, av de styrprogram som kopplingen skall arbeta efter. För att underlätta orienteringen i programlistan finns ett antal REM-satser som innehåller kommentarer. Programmet från sid 00 igenkänns på programraderna 550—650.

```

1 REM *****
2 REM * PROGRAM FÖR AUTOMATISK STYR- *
3 REM * NING AV 9 SWITCHFUNKTIONER *
4 REM * VIA V24 : KONTAKTEN *
5 REM * *
6 REM * 800106 VRSAGYMNASIET ARBOGA *
7 REM * STEFAN ANDERSSON *
8 REM * JAN SMEDBERG *
9 REM *****
10 ; CHR$(12) : DIM A$(500)=8
20 Z=7 AND INP(58)
30 IF Z=7 THEN : CUR(0,0)"KOPPLA IN STYRKORTET" : GOTO 20
40 GOSUB 600 : REM NOLLST
50 ; CHR$(12)"VILL DU SKRIVA ETT STYRPROGRAM (J/N)?" : GET Ø$ : IF Ø$="J" THEN 70
60 ; CHR$(12)"VILL DU HÄMTA ETT STYRPROGRAM (J/N)?" : GET Ø$ : IF Ø$="J" THEN : " JA" : GOTO 730 ELSE ; CHR$(12) : END
70 ; CHR$(12)"ÖNSKAS INSTRUKTIONER? J/N" : GET Ø$ : Ø$ : IF Ø$="J" THEN 230
80 ; "NÄR DATORN SKRIVER UT ETT FRÅGETECKEN"
90 ; "SÅ KAN MAN BESTÄMMA VILKEN AV DE NIO"
100 ; "LYSDIODERNA SOM SKALL TÄNDAS ELLER"

```

```

110 ; "SLÄCKAS. BÖRJA MED ATT TALA OM VILKEN"
120 ; "LYSDIOD DET GÄLLER (A-I). VILL DU HA"
130 ; "FÖRDRÖJNING MELLAN TÄNDNINGARNA RESP."
140 ; "SLÄCKNINGARNA TALA DA OM HUR LANG DEN"
150 ; "SKALL VARA I SEKUNDER."
160 ; ; "***** EXEMPEL *****"
170 ; ; "?A (TÄNDER A)" ; "?B (TÄNDER B)" ; "?10 (VANTAR 10 SEC)" ; "?E (TÄNDER E)" ; "?5 (VANTAR 5 SEC)"
180 ; "?B (SLÄCKER B)" ; "?E (SLÄCKER E)" ; "?25 (VANTAR 25 SEC)" ; "?A (SLÄCKER A)"
190 ; "?R (R SKRIVER DU NÄR DU ÄR FÄRDIG)"
200 ; ; ; "TRYCK PÅ RETURN NÄR DU LAST FÄRDIGT!!";
210 GET G# : IF G#<>CHR$(13) THEN 210
220 REM SKRIVA STRYPPROGRAM
230 ; CHR$(12)"BÖRJA NU!!" : R=0
240 INPUT A$(R)
250 ONERRORGOTO 530
260 IF LEN(A$(R))<1 OR LEN(A$(R))>1 AND ASC(A$(R))>58 THEN ; CHR$(7) : GOTO 240
270 IF A$(R)<>"R" R=R+1 : GOTO 240
280 ; CHR$(12)
290 ; CUR(4,13)"TRYCK PÅ START"
300 POKE 32266,164 : POKE 32284,164
310 REM VANTAR PÅ STARTSIGNAL
320 Z=7 AND INP(58)
330 REM RAD 350 GER EN KORT PULS TILL MOTORSTYRINGSUTGÅNGEN PÅ ABC 80. OM MAN VILL ATT STYRPROGRAMMET SKA REPETERAS
340 REM SÅ KAN MAN KOPPLA EN LEDNING FRÅN DEN TILL STARTKNAPPEN PÅ STYRKORTET OCH SEDAN TA BORT "REM" FRÅN RAD 350
350 REM OUT 58,32 : FOR A=1 TO 80 : Z=7 AND INP(58) : NEXT A : OUT 58,0
360 IF INP(56)=211 THEN GET G# : GOTO 670
370 IF INP(56)=206 THEN GET G# : GOTO 50
380 IF Z<7 THEN 320
390 ; CHR$(12)
400 REM SKRIVER TABELL
410 ; CUR(6,13)"STOPP START"
420 FOR I=1 TO 9 ; CUR(6+I,10)CHR$(I+64) " _ " : A(I)=0 : NEXT I
430 FOR A=1 TO 1000 : NEXT A
440 FOR I=0 TO R-1
450 ONERRORGOTO 530
460 IF ASC(A$(I))<58 THEN 510 : REM HOPP TILL 450 OM A$(I)= EN SIFFRA
470 V=ASC(A$(I))-64 : IF V>9 THEN 500 : REM HOPP TILL 500 OM A$(I) INTE ÄR A-I
480 IF A(V)=0 THEN ; CUR(6+V,15) " _ " : A(V)=1 ELSE ; CUR(6+V,15) " _ " : A(V)=0
490 GOSUB 560
500 NEXT I
510 FOR J=1 TO VAL(A$(I))*1000 : NEXT J : REM FÖRDRÖJNING TID
520 FOR J=1 TO INT(.6*VAL(A$(I))) : NEXT J : GOTO 500 : REM TIDKORREKTION
530 IF I<9 THEN ; CUR(3,10)"FEL I DITT PROGRAM"
540 GOTO 290
550 REM PULSER TILL RAKNARE
560 FOR L=1 TO V
570 OUT 58,16
580 OUT 58,24
590 NEXT L
600 OUT 58,8 : REM PULS TILL LATCH
610 REM NOLLSTÄLLNING AV RAKNARE
620 OUT 58,16
630 Z=7 AND INP(58)
640 IF Z=6 THEN OUT 58,0 : RETURN
650 OUT 58,24 : GOTO 620
660 REM SPARA STYRPROGRAM
670 ; CHR$(12)"GE UTFILNAMN (MAX 8 BOKSTÄVER)"CUR(1,9)_"CUR(0,0) : INPUT C$
680 PREPARE C$+".STR" ASFILE 1
690 FOR I=0 TO R : ; ELA$(I) : NEXT I
700 CLOSE 1
710 GOTO 280
720 REM HÄMTA STYRPROGRAM
730 ; "VAD HETER FILEN" : INPUT C$
740 ONERRORGOTO 800
750 OPEN C$+".STR" ASFILE 1
760 ONERRORGOTO 790
770 R=0
780 INPUT EL A$(R) : R=R+1 : GOTO 780
790 CLOSE 1 : R=R-1 : GOTO 280
800 ; "HITTAR EJ FILEN"CHR$(7) : FOR A=1 TO 2000 : NEXT A : GOTO 20

```


I följande förklaring till programmet är siffrorna inom parentes programradshänvisningar.

Datorn börjar med att undersöka om styrkortet är anslutet till V:24-kontakten (20—30). I en subrutin (600—640) undersöks om dekadräknaren är noll-ställd. Första gången programmet körs finns inget styrprogram att hämta på fil (60) utan ett sådant skall skrivas (50). Instruktioner för skrivandet presenteras på bildskärmen (70—200). När styrprogrammet skrivs (220—280) måste stora bokstäver användas. Skrivs flera bokstäver efter ett frågetecken markerar datorn detta fel genom en akustisk signal och en extra radframmatning. Då styrprogrammet är färdigskrivet markeras detta för datorn genom att trycka på R(270). På skärmen skrivs då TRYCK PÅ START, styrkortets resettangent skall nu tryckas ner varvid datorn börjar köra styrprogrammet.

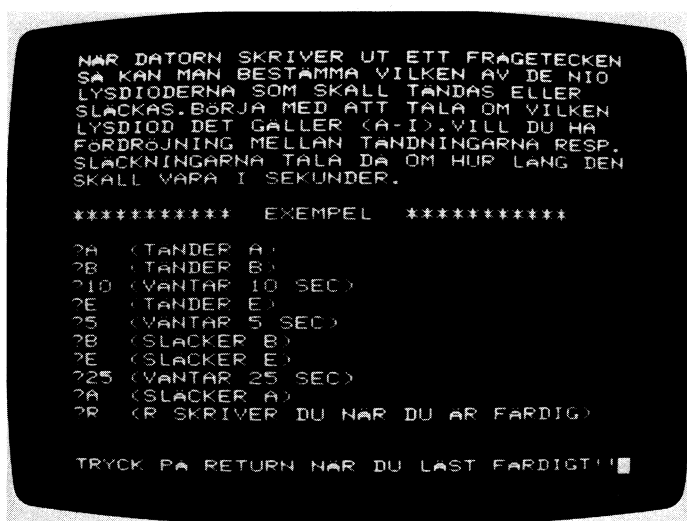


Bild 8. Man kan få instruktioner till hjälp när programmet körs.

Styrfunktionen kan nu följas dels på lysdioderna och dels på skärmen. Skall styrprogrammet köras ytterligare en gång, trycker man bara på resettangenten ännu en gång.

Om styrprogrammet skall lagras på kassett eller flexskiva görs detta genom att trycka på S (660—710). På skärmen skrivs då GE UTFIL NAMN. Skriv då in ett namn på max åtta bokstäver eller siffror. Om programmet t ex ges namnet STYR 80 sker lagringen under namnet STYR80.STR. Räkneverkets inställning bör antecknas om programlagring sker på kassett.

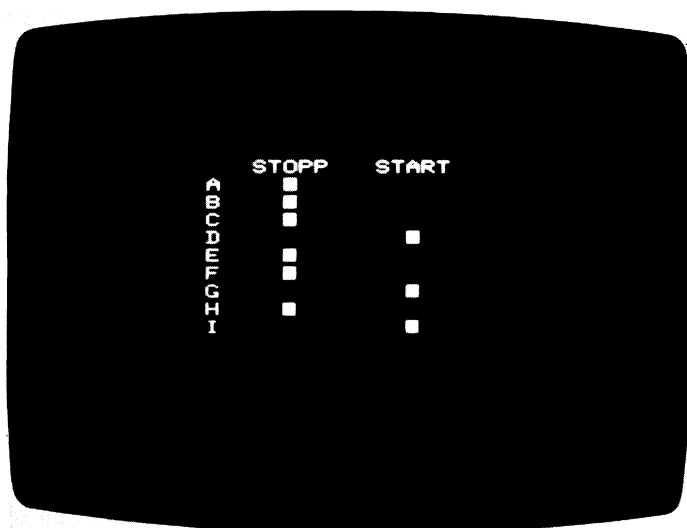


Bild 9. Styrfunktionen kan följas på bildskärmen.

Processkontroll

I ett fjärrmät- eller telemetrisystem matas informationen från ett större eller mindre antal mätpunkter till en dator. Ett sådant system består av en sändardel, ett överföringsmedium och en mottagardel. Sändardelen kan utgöras av givare som omvandlar de fysikaliska storheter som skall mätas till elektriska signaler. Överföringsmediet är den ledning som förbinder givarna med en mottagare som i detta fallet är en dator.

Den koppling som framgår av bild 10 möjliggör kontinuerlig övervakning av åtta avkänningspunkter. En förutsättning är dock att anpassning till mätpunkten har skett med t ex optokopplare, schmitttriggers, komparatorer, gränslägesbrytare o d.

Kopplingen, som styrs av ett program, känner multiplext av status på ingångarna. Om någon ingång noll-ställs (logisk nivå) noteras detta av programmet som stoppar och ger akustiskt larm. Detta larmtillstånd förblir till dess att den ingång som orsakat larmet återställts. När detta har skett återgår programmet automatiskt till avsökningen. I programmet kan valfria kommentarer skrivas in i tabellform så att man direkt ser vilket mätställe som orsakat larmet.

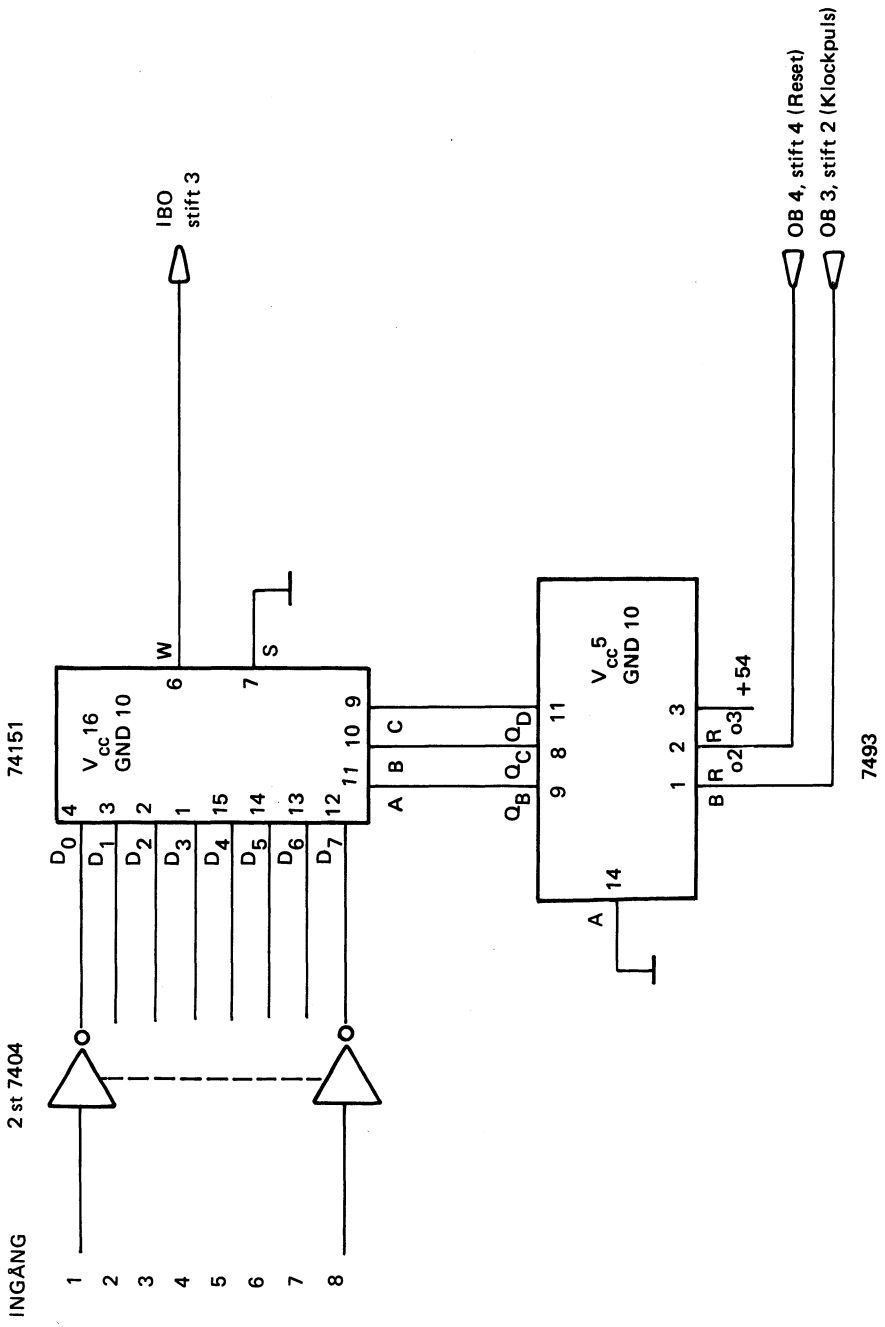


Bild 10. Schema Prozesskontroll.

Skriv in programmet och lagra det på flexskiva eller kassett. När programmet startas visas på skärmen en meny.

- 1.=Inskrivning av ny text i tabell
- 2=Inläsning av tidigare text

Om alternativ 1 väljs skall en ledtext eller rubrik på max åtta bokstäver ges. Denna text t ex SYSTEM 1 används som filnamn vid lagring. Tryck på RETURN och skriv den text som önskas t ex PUMP 1. Efter att på nytt ha tryckt på RETURN skrivs nästa rad. När åtta rader, motsvarande antalet ingångar, är inmatade lagras dessa på kassett/flexskiva med den aktuella rubriken plus en header, i detta falet TXT som står för textfil. I och med att lagringen är klar startar övervakningen direkt. På skärmen skrivs nu texten ut samtidigt som en cursor börjar löpa bredvid texten synkront med avkänningen av mätpunkterna.

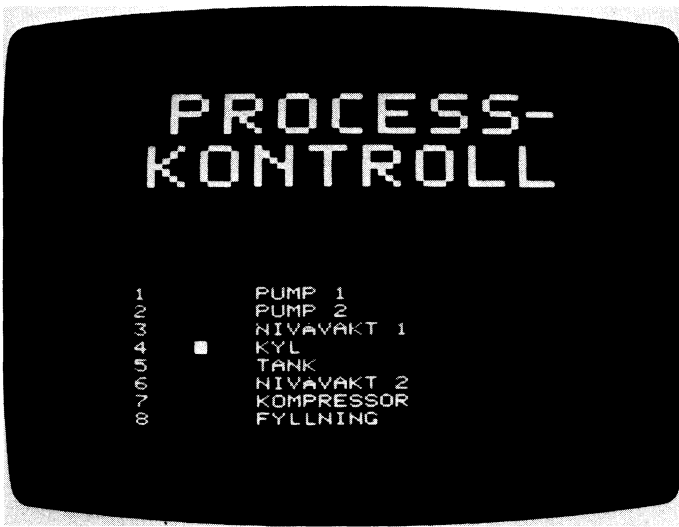


Bild 11. När någon ingång blir nollställd stannar cursorn och markerar den ingång som utlöste larmet.

Pröva nu funktionen hos kortet. Om samtliga ingångar har en nivå som svarar mot en logisk etta skall inget hända. Blir däremot någon ingång noll-ställd (logisk nolla) skall larm ges och cursorn stanna och indikera motsvarande ingång. När vilotillståndet hos ingången återställts skall övervakningen fortsätta. Om alternativ 2 på menyn väljs används tidigare skriven textfil. Ange

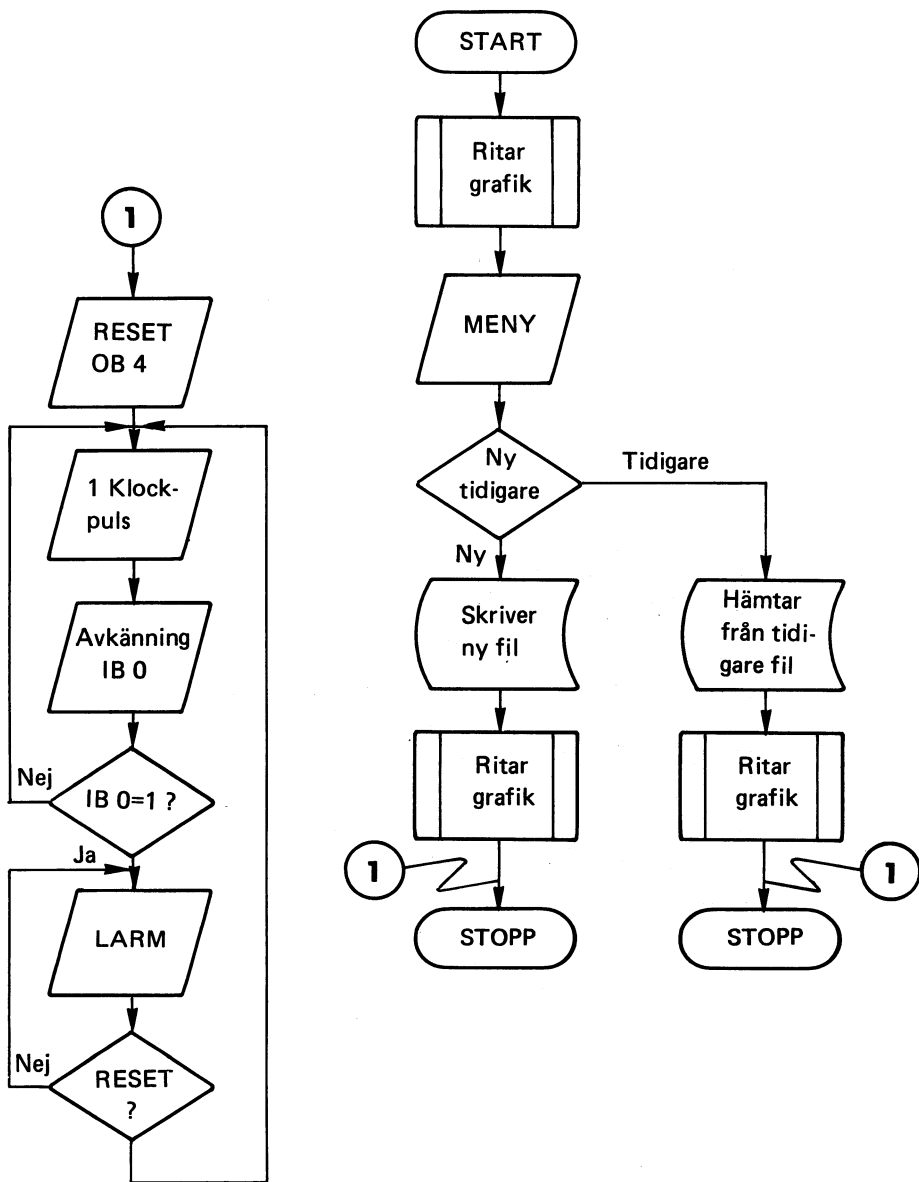


Bild 12. Flödesschema för programmet "Processkontroll".

textfilens namn och tryck på RETURN. Textfilens innehåll skrivs nu ut på skärmen och övervakningen startar direkt. I ett system av detta slag är det inte bara den hårdvarubundna konstruktionen som måste ha en hög driftsäkerhet. Utöver avlusning av bugar har programmet försetts med en felhanteringsrutin (error handler) som ligger på programraderna 108—128. Följande felmeddelande ges av programmet.

- >Felaktigt tal=ogiltig inmatning t ex siffror först i ett filnamn
- >Kan ej öppna fler filer. Gäller endast flexskiva
- >Hittar ej filen
- >Slut på filen=Filen slut innan läsningen är klar
- >Checksummafel vid läsning
- >Checksummafel vid skrivning
- >Recordnummer utanför filen. Gäller endast flexskiva
- >Filen skrivskyddad. Gäller endast flexskiva
- >Filen raderingsskyddad. Gäller endast flexskiva
- >Skivan full
- >Skivan ej klar. Formatering?
- >Skivan skrivskyddad

```

10 ; CHR$(12)
11 REM *****
12 REM * *
13 REM * PROCESS - *
14 REM * *
15 REM * KONTROLL *
16 REM * *
17 REM * LARS KJELLGREN & *
18 REM * ANDERS PALM *
19 REM * VASAGYMNASIET *
20 REM * ARBOGA 80-03-24 *
21 REM * VER 1.0 *
22 REM *****
23 REM * * * CROSS-REFERENCE * * *
24 REM P$=MENYVAL
25 REM F$=NAMN PÅ FILEN
26 REM X=VARVRÅKNARE, STRÅNGPOS, INDEX FÖR A$
27 REM A$(X),A$=KOMMENTARER & GRAFIK
28 REM I=VÄNTELOOP
29 REM P=PULSGENERATOR, CURSORPOS.
30 REM D=ERRCODE (DATA)
31 REM G=ERRCODE
32 REM E$=ERRORMESSAGE (DATA)
33 REM
34 GOSUB 98 : REM RITAR GRAFIK
35 REM
36 REM * * * * M E N Y * * *
37 REM
38 ; CUR(8,18)"MENY"CUR(9,18)"____"
39 ; CUR(11,0)"1=INSKRIVNING AV NY TEXT I TABELL"
40 ; CUR(13,0)"2=INLÄSNING AV TIDIGARE TABELLTEXT" : ; ; "VALJ 1 ELLER 2" : GET P$
41 IF P$<"1" OR P$>"2" THEN 39

```



```

42 ON VAL(P#) GOTO 46,65
43 REM
44 REM * * * * NY TEXT * * *
45 REM
46 ; CHR$(12)
47 ; "NAMNGE DEN NYA TEXTEN (MAX 8 TECKEN)" ; ; " (SE TILL ATT TILLGÅNGLIGT UTRYMME" ; ; "FINNS PÅ SKIVAN/KASSETTEN)"
48 INPUT F# : IF LEN(F#)>8 OR LEN(F#)<1 THEN 47
49 ; CHR$(12)CUR(0,16)F#
50 FOR X=11 TO 18 : ; CUR(1,0)SPACE$(80)CUR(1,0)
51 ; X-10 : ONERRORGOTO 111 : INPUTLINE A# : A#=LEFT$(A#,LEN(A#)-2)
52 A#(X-10)=CHR$(27,61,X+32,14+32)+A#
53 ; A#(X-10)
54 NEXT X
55 ONERRORGOTO 109
56 PREPARE F#+".TXT" ASFILE 1
57 FOR X=1 TO 8
58 ONERRORGOTO 109
59 ; £1A$(X)
60 NEXT X
61 CLOSE 1 : GOSUB 98 : GOTO 76 : REM RITAR FÖRST GRAFIK, GÅR SEDAN TILL AVSNITT PROCESSKONTROLL
62 REM
63 REM * * * TIDIGARE TEXT * * *
64 REM
65 ; CUR(8,0)SPACE$(340)CUR(8,0)"ANGE TEXTENS NAMN" : INPUT F#
66 ; CUR(8,0)SPACE$(560)
67 REM ONERRORGOTO 520
68 OPEN F#+".TXT" ASFILE 1
69 FOR X=1 TO 8
70 ONERRORGOTO 109 : INPUTLINE £1A$(X)
71 NEXT X
72 CLOSE 1 : GOSUB 98 : GOTO 76 : REM RITAR FÖRST GRAFIK, GÅR SEDAN TILL AVSNITT PROCESSKONTROLL
73 REM
74 REM * * * UTSKRIFT * * *
75 REM
76 FOR X=1 TO 8 : ; A$(X)
77 NEXT X
78 REM
79 REM * * * PROCESSKONTROLL * * *
80 REM
81 OUT 58,247% AND INP(58) : REM NOLLSTÄLLER OB 3 PÅ PIO
82 OUT 58,239% AND INP(58) : REM NOLLSTÄLLER OB 4 PÅ PIO
83 FOR I=0 TO 100 : NEXT I
84 OUT 58,16% OR INP(58) : REM ETTSTÄLLER OB 4 PÅ PIO
85 FOR P=11 TO 18
86 OUT 6,0 : REM STOPP AV LJUD
87 OUT 58,247% AND INP(58) : REM NOLLSTÄLL OB 3 PÅ PIO
88 OUT 58,8% OR INP(58) : REM ETTSTÄLL OB 3 PÅ PIO
89 ; CUR(P,5)P-10;CUR(P,10)"_"
90 FOR I=0 TO 25 : NEXT I : REM ANPASSAR CURSORNS RÖRELSE TILL MONITORS BILD- FREKVENNS
91 ; CUR(P,10)CHR$(32)
92 IF INP(58) AND 1% THEN NEXT P ELSE 94 : REM TESTAR UTGÅNG PÅ MULTIPLEXER
93 GOTO 85
94 ; CUR(P,10)"_" : OUT 6,251 : REM GER LARM
95 IF NOT (INP(58) AND 1%) THEN 92 ELSE 95 : REM TESTAR OM LARMET ÄR ÅTERSTÄLLT
96 END
97 REM
98 REM * * * RITAR GRAFIK * * *
99 RESTORE 101
100 ; CHR$(12) : FOR I=0 TO 5 : ONERRORGOTO 111 : READ A# : ; CUR(I,0);CHR$(151);A# : NEXT I
101 DATA " JEI 7E4HEI 6E 7E!HEE 6E!"
102 DATA " JE! 7G J J 5 7E EI ""E4EE"
103 DATA " "" ! ! E! ""E EE!""E! EE"
104 DATA " 5B!HEI =0S""KE 7E4HEI 5 J"
105 DATA " 7D J J 5""5 J 7G J J 5 J"
106 DATA " ! ! E! ! ! "" ! ! E! EE!""EE"
107 RETURN
108 REM ERRORHANDLER
109 CLOSE 1
110 RESTORE 117
111 READ D,E# : G=ERRCODE

```

```

112 IF D=G THEN 115 ELSE 111
113 IF D=0 THEN 34 ELSE 34
114 REM GOTO 929
115 ; CHR$(12)CUR(12,10)D,E$
116 FOR I=0 TO 5000 : NEXT I : G=0 : GOTO 34
117 DATA 12, "FELAKTIGT TAL"
118 DATA 19, "KAN EJ ÖPPNA FLER FILER"
119 DATA 21, "HITTAR EJ FILEN"
120 DATA 34, "SLUT PÅ FILEN"
121 DATA 35, "CHECKSUMMAFEL VID LÄSNING"
122 DATA 36, "CHECKSUMMAFEL VID SKRIVNING"
123 DATA 38, "RECORDNUMMER UTANFÖR FILEN"
124 DATA 39, "FILEN SKRIVSKYDDAD"
125 DATA 41, "SKIVAN ÖNWKTER"
126 DATA 42, "SKIVAN EJ KLAR"
127 DATA 43, "SKIVAN SKRIVSKYDDAD"
128 DATA 0, "SLUT"

```

Logik monitor

De tidigare beskrivna V:24-applikationerna har utgjort exempel på hur ABC80 kan användas för att styra funktioner via V:24-snittet. Med hjälp av detta snitt kan även mätapplikationer förverkligas, exempelvis mätning av pulsfrekvens och pulslängd. Med den koppling som framgår av bild 13 kan ABC80 användas som logikclips. En dyr sådan, men en vidareutveckling av denna teknik öppnar nya möjligheter framförallt vid digital felsökning. Att låta datorn felsöka blir en realitet i framtiden.

Kopplingens uppgift är att mäta samt på bildskärmen presentera de logiska nivåerna på en 14- eller 16-pinnars IC. Av presentationen skall också framgå om det på något stift ligger ett pulståg. Kopplingen är försedd med 16 stycken mätängångar A—P. Till dessa är anslutet en 16-ledars flatkabel som i andra ändan är försedd med en IC-klämma. Den hårdvara som denna koppling utgör, måste för att fungera kompletteras med ett styrprogram. Vid exekveringen av detta väljer man om mätning skall göras på 14 eller 16-pinnars IC.

Varje mätängång avsöks av en multiplexer (74154). Avsökningshastigheten, dvs den tid som mätningen på varje ben får ta, kan i programmet varieras. För att datorn skall veta om det är en stabil nivå eller en puls får tiden inte vara för kort. På bildskärmen skall markeringen P betyda att motsvarande IC-ben har en puls. Om pulsfrekvensen är en Hz fordras en mättid av ca 0,6 sekunder för att med säkerhet registrera ett omslag. Är pulsfrekvensen 100 Hz blir periodtiden 10 ms varför mättiden bör vara 6 ms. Under mätningen kan man på skärmen se vilket IC-ben som avsöks. Detta markeras ovanför respektive under varje ben av ett minustecken som utgör markör. Hastigheten som denna rör sig med är då en indikation på mättiden. Genom programkonstruktionen kommer ingen utskrift att göras på det första varvet.

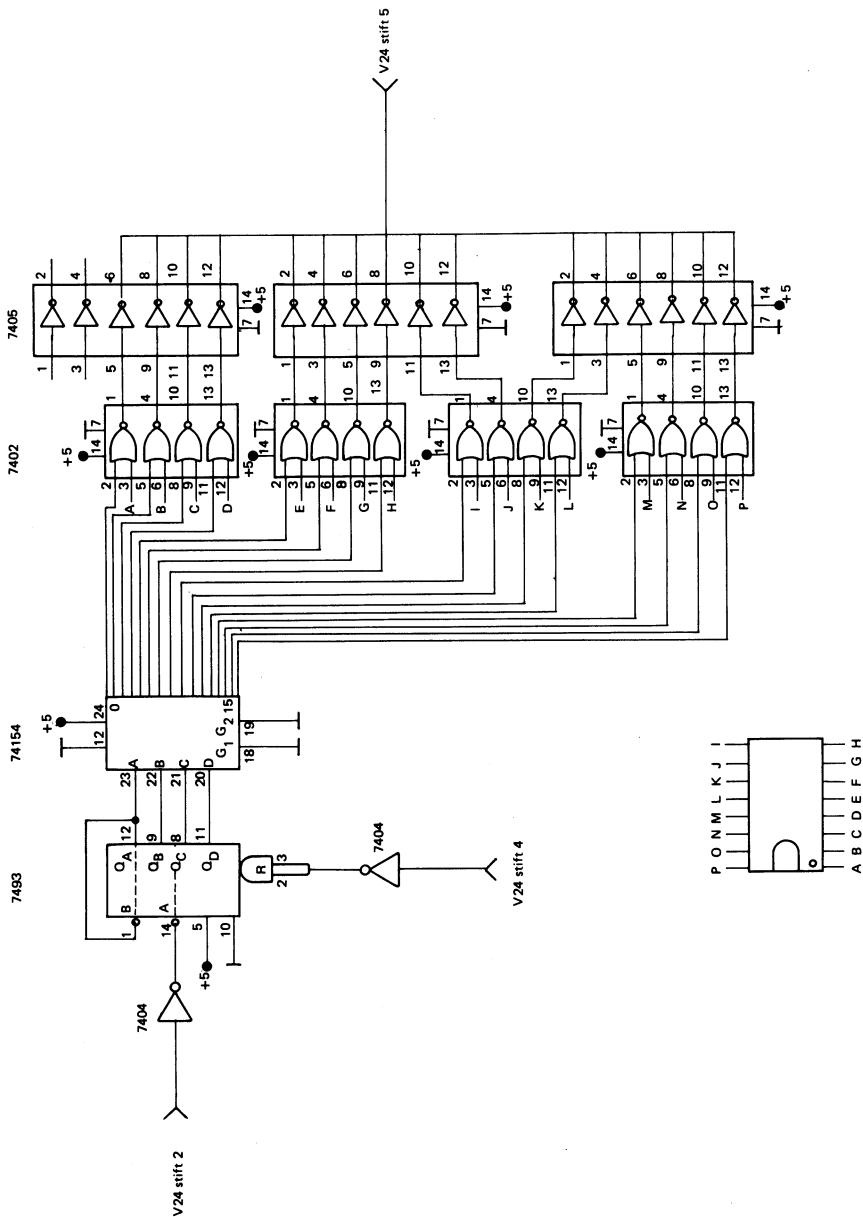


Bild 13. Schema Logikmonitor.

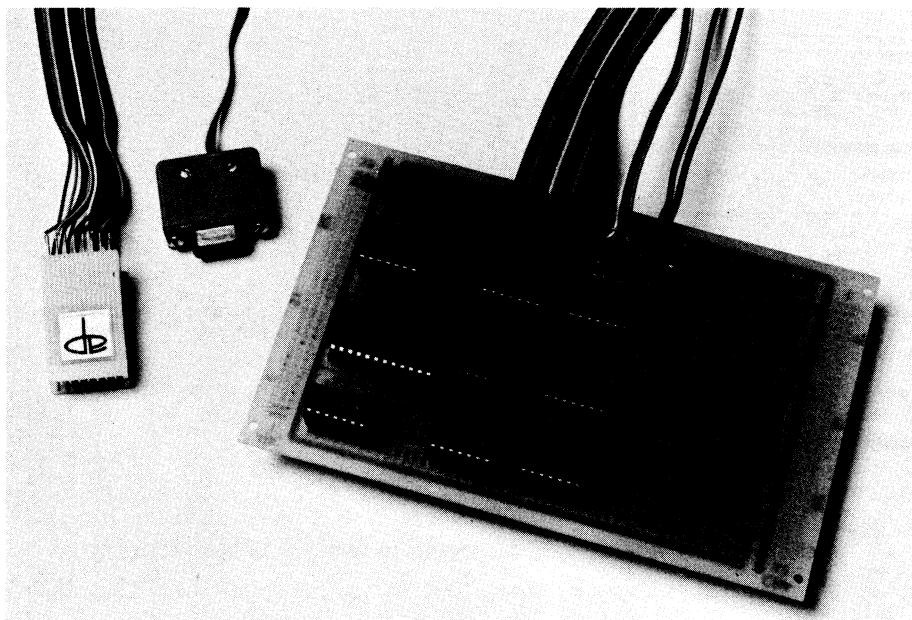


Bild 14. Logikmonitorn är här uppbyggd på standard Europakort

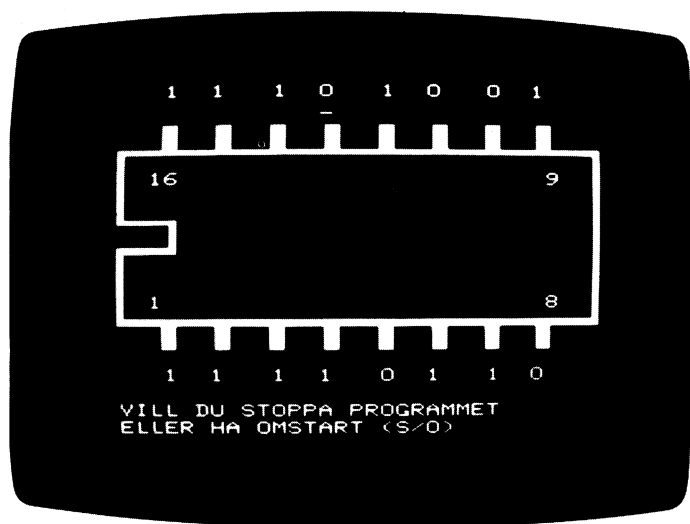


Bild 15. På bildskärmen kan avsökningen av IC:n ben följas. Avsökningshastigheten kan variera beroende på mätobjektets pulsfrekvens. Stiftindikering kan ske med 1, 0 och P där P står för puls.

```

1 REM *****
2 REM *
3 REM * LOGIC MONITOR *
4 REM *
5 REM * JAN SMEDBERG & ANDERS PALM *
6 REM *
7 REM * VRSAGYMNASIET *
8 REM *
9 REM * 1988. 01. 17 ARBOGA *
10 REM *
11 REM *****
12 DIM X$(500)
13 ; CHR$(12)
14 ; CUR(5,7)*"LOGIC MONITOR"
15 ; CUR(6,7)*" "
16 ; "DETTA PROGRAM KAN MED HJALP AV V24-" ; "KONTRAKTEN OCH EN SPECIAL UPPKOPLING!" ; "MATA VAD DU HAR FOR NIVAER PA EN "
17 ; "14- ELLER EN 16-PINNARS IC-KRETS." ; "DU KAN AVEN MATA PULSER." ; "(TRYCK PA RETURN)";
18 GET X$ ; CHR$(12)
19 REM
20 REM ***** MENY!!!!!! *****
21 REM
22 ; CUR(5,2)*"***** MENY *****"
23 ; CUR(7,5)*"1 MATNING PA 14-PIN'S"
24 ; ; TAB(5)*"2 MATNING PA 16-PIN'S"
25 ; ; TAB(5)*"ANGE MATNING (1/2)";
26 ONERRORGOTO 22 : GET X$ : IF VAL(X$)<1 OR VAL(X$)>2 THEN 22
27 ; CHR$(12) : ONERRORGOTO 27
28 ; ; "VILKEN FREKVENNS AR DEN LAGSTA DU" ; "TROR KAN FOREKOMMA I DIN MATUPP-" ; "KOPPLING (ANGE DEN I HZ)"
29 INPUT E : D=200/E : IF D<10 THEN D=10
30 OUT 58,16 : REM NOLLSTALLER SIFT 4(RAKNAREN)
31 A%=1
32 ; CHR$(12)
33 ON VAL(X$) GOSUB 70,97
34 REM
35 REM ***** MATER PA PINNARNA *****
36 REM
37 A%=4% : A%=4%
38 FOR X%=1 TO P% STEP 35
39 IF INP(56)=211 OR INP(56)=243 THEN 121
40 IF INP(56)=207 OR INP(56)=239 THEN 18
41 ; CUR(A%*(X%/10)+8)*"_"
42 IF A%=1 THEN 46 ELSE GOSUB 58
43 IF B$="P" THEN 45
44 IF A%=1 THEN 46 ELSE Z%=7% AND INP(58) : IF Z%=5 THEN B$="1" ELSE B$="0"
45 ; CUR(A%-1*(X%/10)+8)B$
46 OUT 58,8 : REM START AV KLOCKPULS
47 OUT 58,247% AND INP(58) : REM STOPP AV KLOCKPULS
48 ; CUR(4%*(X%/10)+8)SPACE$(39%) ; ; CUR(18%*(X%/10)+8)SPACE$(39%) ; ; CUR(28%*(X%/10)+8)SPACE$(39%)
49 NEXT X%
50 IF A%=20 THEN 52 ELSE 51
51 A%=A%+14% : A%=A%+14% : GOTO 38
52 OUT 58,16 : REM NOLLSTALLER STIFT 4(RAKNAREN)
53 A%=0
54 GOTO 37
55 REM
56 REM ***** KOLLAR OM PULS *****
57 REM
58 U=0 : B$=""
59 U=1+U : X$(U)=INP(58)
60 IF U<D THEN 59
61 FOR S%=1 TO D-1
62 IF X$(S%)<>X$(S%+1) THEN 65 ELSE 63
63 NEXT S%
64 RETURN
65 B$="P" : RETURN
66 END
67 REM
68 REM ***** RITAR 14 PIN'S IC *****
69 REM
70 ; CHR$(12)

```




ABC-bussen

På datorenhetens baksida sitter en 64-polig DIN-kontakt (41612), genom vilken man kommer i kontakt med ABC-bussen. Denna är i väsentliga delar kompatibel med den s k 4680-bussen. ABC-bussen används, liksom V:24-snittet, för att expandera ABC80. Det vanligaste sättet att använda ABC-bussen är till anslutning av printar och flexskiveminne, men bussen används också för minnesexpansion samt för anslutning av I/O-kort som gör det möjligt att använda ABC80 som styr- och mät dator. Bussen är, som framgår av bild 16,

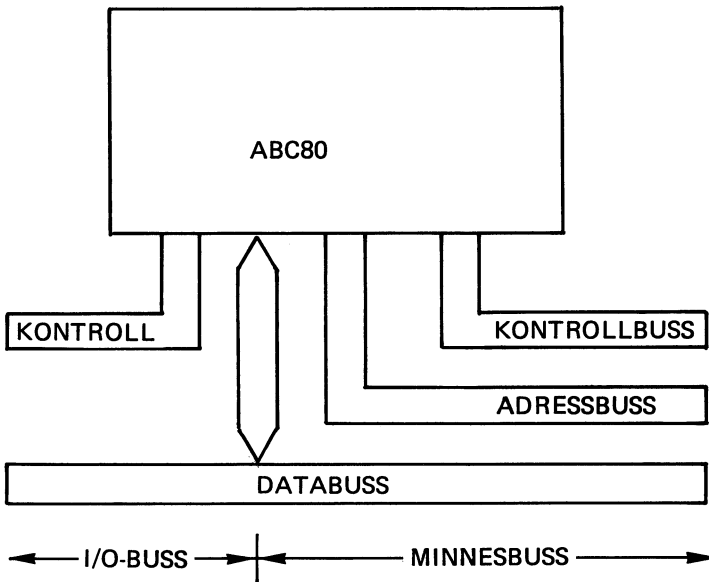


Bild 16. ABC-bussens struktur.

uppdelad i en minnesdel och en I/O-del. Stiftkonfigurationen för ABC-busskontakten framgår av bild 17. Till stiften a6—a13 är den dubbelriktade databussen ansluten. Kontrollbussen (stift a15—23) är fördelad på tre inportar (stift a15—17) samt sex utportar (stift a18—23). Adressbussen är åtkomlig på stift b14—b29. Till minnesbussen, bestående av kontroll-adress- och databussarna, kan anslutas RAM- eller PROM-kort. Till I/O-bussen, bestående av kontroll- och databussarna, kan upptill 64 I/O-kort anslutas. I/O-korten kan utgöras av interface eller styrkort. Kortval sker med signalen CS (card select) stift a23.

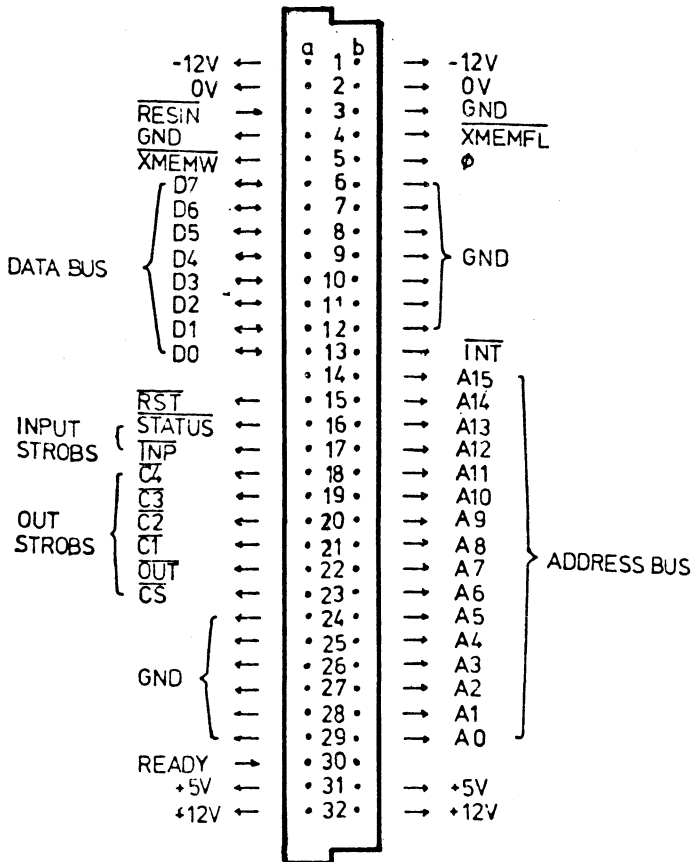


Bild 17. ABC-bussens stiftkonfiguration.

Max belastning på bussen

+5V 250 mA

+12V 125 mA

-12V 75 mA

±12 volt går även till v:24-kontakten varför det totala kraftuttaget inte får överstiga värdena i tabellen ovan. Alla IN/UT-signaler är TTL-kompatibla.

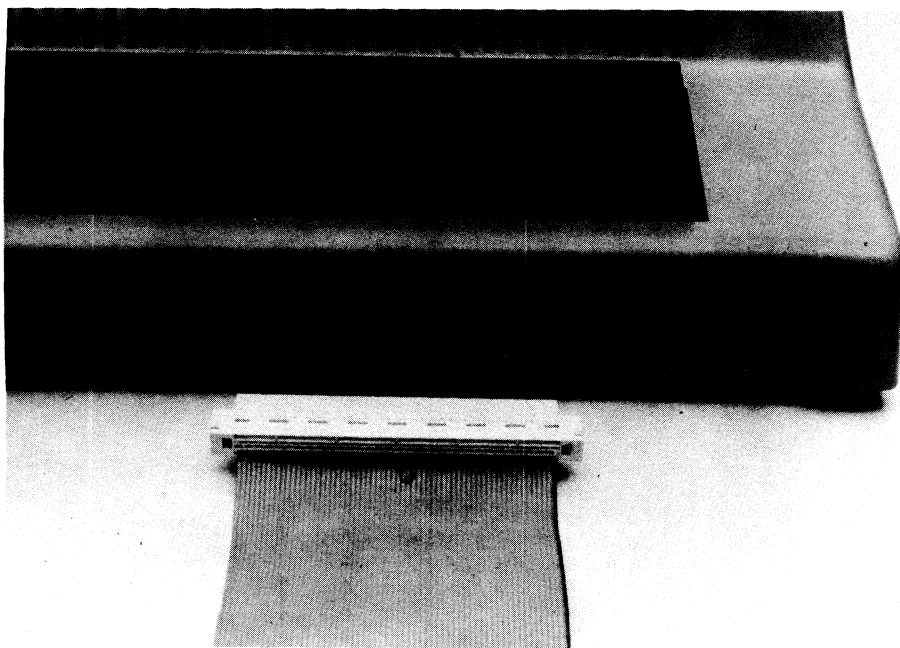


Bild 18. Datorenhetens busskontakt.

Kommandon för kommunikation på ABC-bussen är:

	PORT	BASIC	FUNKTION
OUT	UT 0	OUT	DATA UT (OUT)
	UT 1	OUT 1	KORTVAL (CS)
	UT 2	OUT 2	C1 KOMMANDO (C1)
	UT 3	OUT 3	C2 KOMMANDO (C2)
	UT 4	OUT 4	C3 KOMMANDO (C3)
	UT 5	OUT 5	C4 KOMMANDO (C4)
IN	IN 0	INP(0)	DATA IN (INP)
	IN 1	INP(1)	STATUS IN (STATUS)
	IN 7	INP(7)	RESET (RST)

Expansionsmöjligheter med ABC80



Bild 19a. Flexskiveminnet DataDisc 80 (DD80) har samma utseende som DataDisc 82.

Ett flexskiveminne (floppydisc) ger, förutom god lagringskapacitet och kort åtkomsttid, i regel också ett expansionsutrymme för minnes- och I/O-expansion. Flexskiveenhet ABC, FD2, DataDisc 80 och 82 samt DataDisc 88 är exempel på sådana. Flexskiveenhet ABC, FD2 och DataDisc 80 resp 82 är 5,25" flexskiveminnen. Lagringskapaciteten är normalt 320 sektorer/flexskiva. En sektor rymmer 256 byte varför den totala lagringskapaciteten blir 81920 byte. Normalt räknar man dock med att endast 296 sektorer är tillgängliga för användaren. Lagringskapaciteten blir därför ca 75K byte. Med "double density", som flexskiveenhet ABC, DataDisc 82 och FD2-D har, ökar kapaciteten till det dubbla dvs 160K byte/flexskiva. DataDisc 88 är en 8" flexskiveminne med en lagringskapacitet av ca 1M byte/flexskiva.

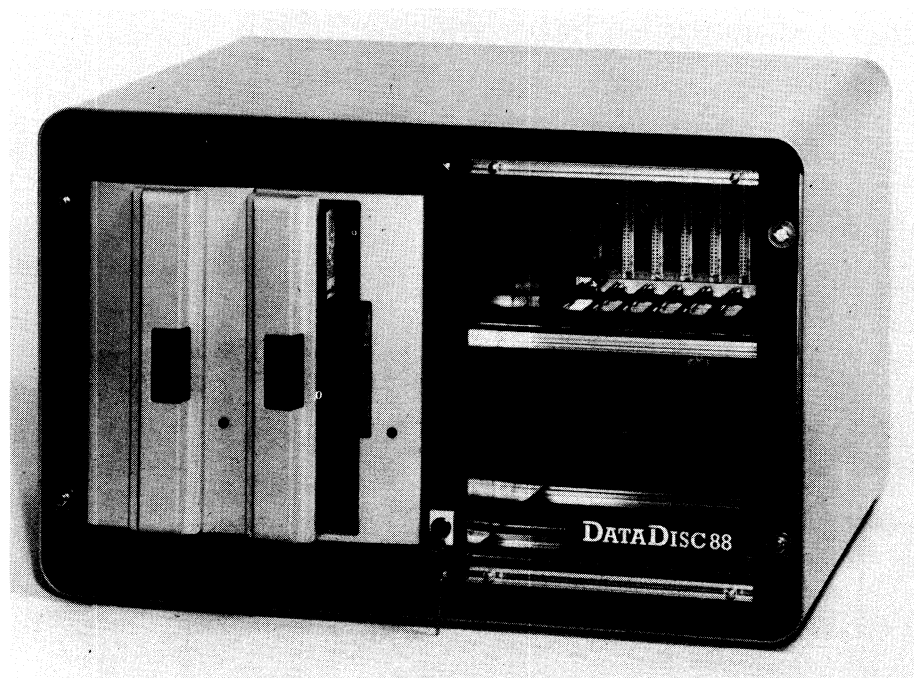


Bild 19b. 8" flexskiveminnet DataDisc 88.

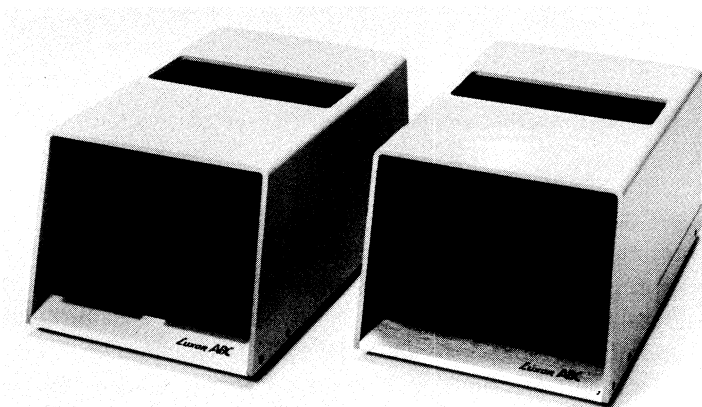


Bild 19c. Luxors flexskiveminne ABC med dubbel packningstäthet, dvs 2*160K byte. Omkoppling mellan enkel och dubbel packningstäthet kan ske med en omkopplare. Flexskiveminnet består av en driveenhet och en expansionsenhet. I den senare finns ett expansionsutrymme för fyra I/O-kort och tre minneskort ur 4680-serien. Genom bygling på bakplanet kan de fyra I/O-kortplatserna lätt anpassas till ABC-kortserien.



Bild 20. Flexskiveminnet FD2.

I dessa flexskiveminnen finns minnes- och I/O-expansionsutrymme. Detta ger bl a den fördelen att expansionskortet får sin strömförsörjning från flexskiveminnet och kommer därför inte att belasta ABC80. I FD2 finns två kortplatser för antingen minnes- eller I/O-expansion. Båda kortplatserna har tillgång till såväl kontroll-, adress- och databussarna. I/O- och minneskort kan därför placeras på valfri plats eftersom I/O-kortet i regel inte har någon adressbuss framdragen. I flexskiveenhet ABC och DataDisc 80 finns expansionsutrymme för åtta kort. Kortplatserna är fördelade på:

- 4 I/O
- 3 Minne
- 1 CPU (bussanslutning från ABC80)

Av dessa är redan tre platser upptagna. Flexskiveminnets controllerkort tar en kortplats i I/O-utrymmet. Kortplatsen längst till höger är upptagen av ett PROM-kort. På detta finns åtta kretshållare för EPROM 2708. Fyra av

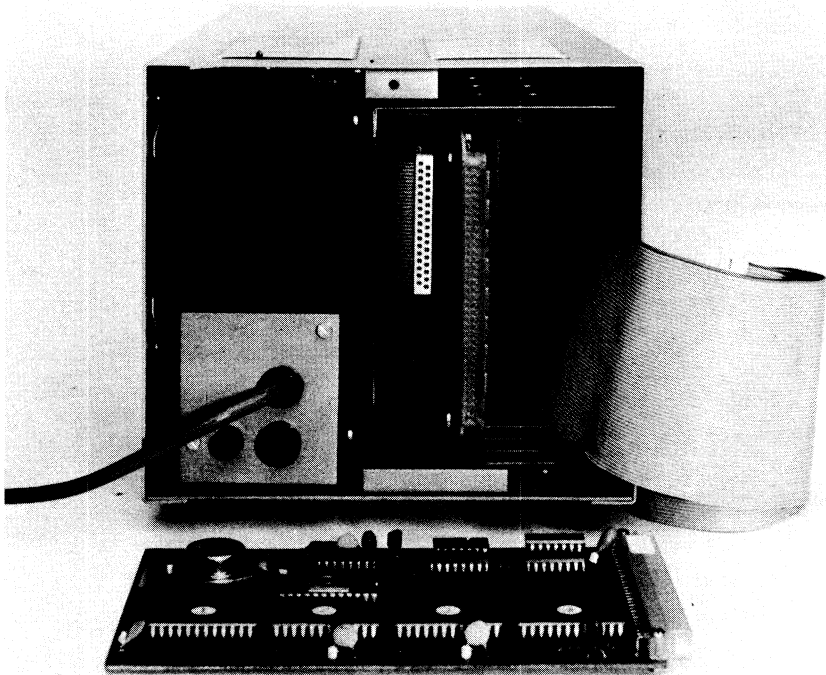


Bild 21. FD2:s expansionsutrymme.

kretshållarna är upptagna av Disc Operativ Systemet (DOS). Övriga fyra platser är avsedda för drivprogram

På den fjärde kortplatsen från höger ansluts busskabeln från ABC80. Till höger om denna anslutning finns nu plats för två minneskort. Till vänster om ABC80 anslutningen finns expansionsutrymme för tre I/O-kort. Ytterligare expansionsutrymme kan erhållas genom komplettering med en separat expansionslåda. Denna kopplas till ABC80:a ABC-busskontakt via en bussförgrening.

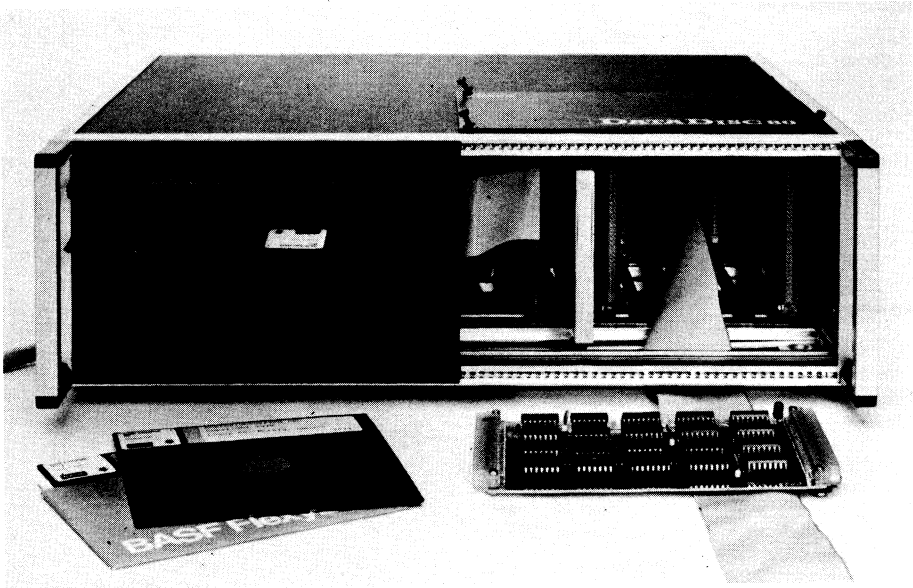


Bild 22. DD80:s alt DD82:s expansionsutrymme.

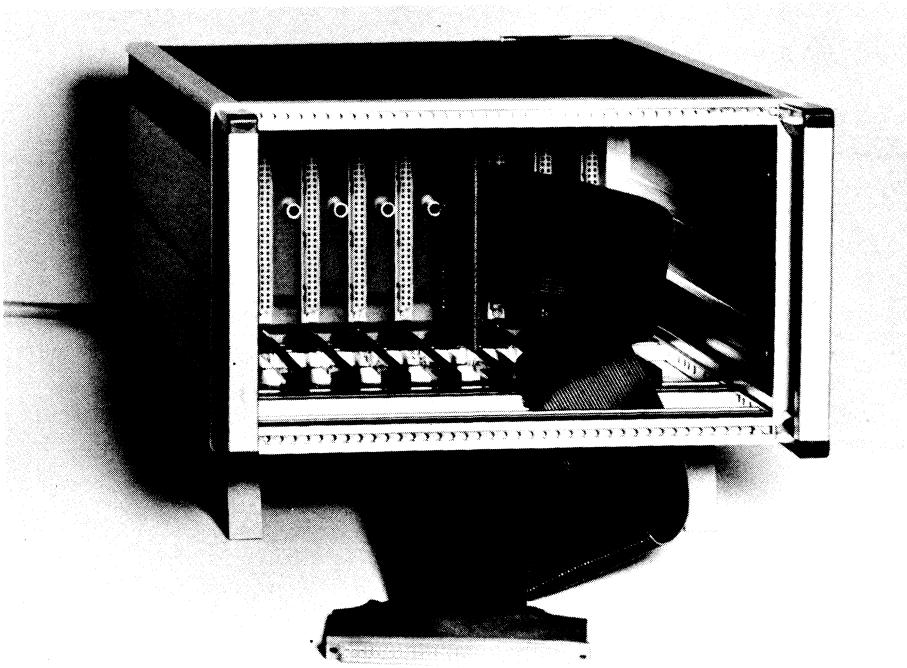


Bild 23. Expansionslåda.

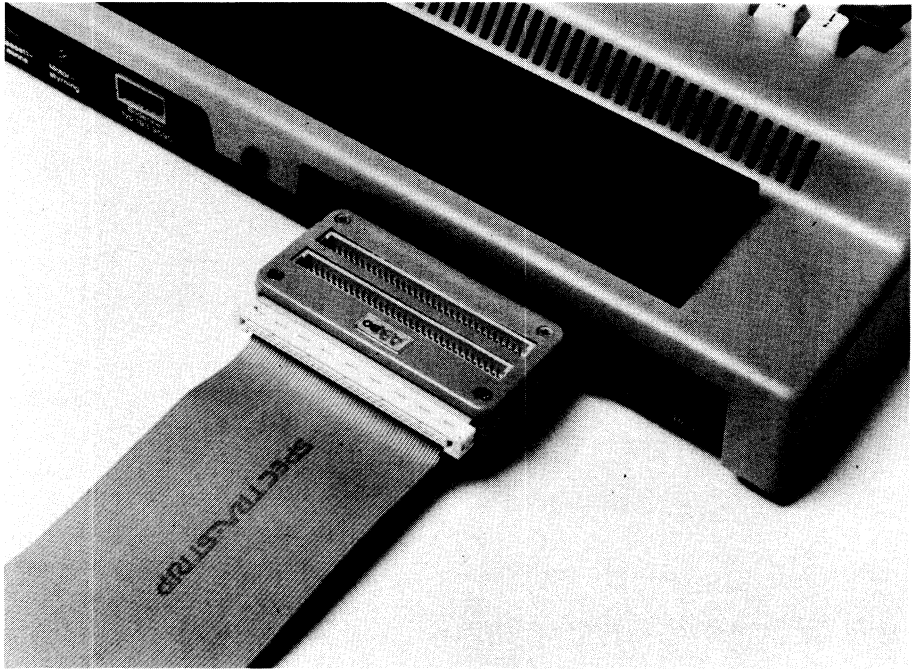


Bild 24. Bussföregreningsdosa.

ABC80 — minne och minnesexpansion

Minnesstacken

Mikroprocessorn i ABC80, Z80A, har en 16-bits adressbuss, vilket begränsar ABC80:s minnesarea till 64K byte fördelade på 32K byte RAM (RWM) och 32K byte ROM. Överst i minnesstacken, vilken framgår av bild 25, ligger användar- eller arbetsminnet. Detta utgör stackens RAM-area. ABC80:s interna arbetsminne är dock bara på 16K byte. Vill man utöka användarminnet till 32K byte kan detta ske genom extern expansion med 16K byte RAM. Normalt sker detta genom att man använder två stycken RAM-kort om vardera 8K byte eller ett dynamiskt RAM-kort på 16K byte. Dessa ansluts, via ABC-bussen, till ett reserverat kortutrymme i flexskiveminnets expansionsdel. Den nedre delen av minnesstacken, ROM-arean, är också fördelad på 16+16K byte med undantag för bildminnet. Det nedersta blocket på 16K byte ROM utgör basictolken eller interpretatorn. Den basicdialekt som används i ABC80 är DATABOARD-BASIC vilken kan betraktas som mycket kraftfull. Ovanför basictolken finns ett reserverat utrymme på 8K byte. Detta gör det möjligt att expandera basictolken till 24K byte. Den del av ROM-arean som nu återstår är avsedd för systemexpansion. De drivprogram som kan vara aktuella är normalt hårdvarubundna i PROM-kapslar. Exempelvis är DOS-programmet (Disc Operativ System) lagrat i fyra stycken PROM-kapslar om vardera 1K byte och placerade i flexskiveminnet.

Som tidigare konstaterats är det interna RAM:et på 16K byte. Vid programskrivning respektive programlagring disponeras dock endast utrymmet mellan adress 49152(dec) och 64256(dec). Skriv

```
PRINT PEEK (65064)
```

ABC80 svarar då med att skriva 251, vilket är det antal block om vardera 256 byte, som utgör det s k "taket". Om flexskiveminne är inkopplat kommer "taket" att ligga 6 block lägre, dvs antalet block blir då 245. Detta beror på att det längst upp i minnesstacken ligger en DOS-buffert som är 8 block stor medan motsvarande kassettbuffert endast omfattar 2 block. Om man får ERR då man laddar in ett program från flexskiva kan "taket" höjas genom att skriva

```
POKE 65064,251
```

RAM-utrymmet ökar nu från 245 till 251 block vilket är lika med 1.5K byte.

ABC80:s MINNESKARTA

Basadress K-Byte	Dec.adress		Hex.adress
	65535		FFFF
		RAM internt användarminne	
48	49152	16 KB	C000
		RAM externt användarminne	
40	40960	16 KB	A000
32	32768	16 KB	8000
		RAM internt Bildminne	
31	31744	1 KB	7C00
		ROM externt drivrutin f. printer	
30	30720	1 KB	7800
		ROM eller RAM externt	
29	29696	1 KB	7400
		ROM externt drivrutin f. IEC	
28	28672	1 KB	7900
		ROM externt DOS Disc Operativ System	
24	24576	4 KB	6000
		ROM internt för expansion	
16	16384	8 KB	4000
		ROM internt BASIC-tolk	
0	0	16 KB	0000

Bild 25. ABC80:s minneskarta.

Följande program ger utskrift av "golvet's" och "takets" adress samt det totala RAM-utrymme som finns. Skriv in och kör programmet. Höj "taket" genom POKE:ning samt kör programmet på nytt. Glöm inte att göra reset efteråt.

```
1 ; CHR$(12,10,10,10)
10 ; "GOLVETS ADRESS ="; : A=PEEK(65053)*256+PEEK(65052) : ; A
20 ; "TAKETS ADRESS ="; : B=PEEK(65064)*256+PEEK(65063) : ; B
30 ; "ANVÄNDBART RAM-UTRYMME ="B-A" BYTES"
```



Bild 26. RAM-undersökning.

Om man vill göra en undersökning av innehållet på de enskilda minnesadresserna kan följande program användas. Detta ger på printer utskrift av det binära innehållet i de minnesceller som ligger mellan de start och stoppadresser man själv väljer. Adresserna kan ligga var som helst i minnesstacken. Om printern som används är ansluten till V:24-kontakten måste programrad 10 ändras.

```
1 OPEN "PR:" ASFILE 1
10 ; CHR$(12)
20 ; "START,STOPP"; : INPUT C,D
30 FOR E=C TO D STEP 5
40 A=PEEK(E)
45 FOR F=0 TO 4
46 A=A+F
```

```

50 ; £1.E+F" ";
60 C=128
80 FOR I=7 TO 0 STEP -1
90 IF A AND C THEN B(I)=1 ; £1" 1"; ELSE B(I)=0 ; £1" 0";
100 C=C/2
110 NEXT I
120 ; £1" ";
130 NEXT F
140 ; £1
150 NEXT E

```

När man skrivit eller kört ett program, undrar man ofta hur mycket av användarminnet (RAM) som programmet upptar. Länka, med hjälp av MERGE-kommandot, följande programrad till det program som skall storleksbestämmas. Programraden ger utskrift av programmets storlek samt hur mycket RAM-area som återstår.

```

1 A=PEEK(65064)*256+PEEK(65063)-PEEK(65057)*256-PEEK(65056) ; ; "KVAR "A" BYTES"
; ; "PROGRAM "13391-A" BYTES" : GET 08 : A=0

```

Minnesexpansion

Som redan konstaterats kan arbets- eller användarminnet (RAM) byggas ut till 32K byte. I flexskiveminne ABC och DataDisc 80 finns det två kortplatser reserverade för minnesexpansion. Minneskort 2055, som är ett 8K byte RAM-kort används. Om man använder flexskiveminnet FD2 kan minnesexpansion till 32K byte också erhållas. Men då måste ett 8K byte RAM-kort med beteckningen ABC RAM användas. Med ett 8K byte minneskort ges en expansion till 24K byte medan två minneskort ger 32K byte RAM. Man kan också expandera direkt till 32K byte RAM. Detta möjliggörs genom ett 16K byte dynamiskt RAM-kort.

För att ABC80 skall upptäcka minnesutökningen krävs att kortadressen ligger omedelbart under ABC80:s eget RAM-minne. Basadressen för det interna RAM:et är 48K byte. Om minnesexpansion skall ske med ett minneskort på 8K byte måste detta alltså få basadressen 40K byte ($48-8=40$). Skall ytterligare expansion göras blir det andra minneskortets basadress 32K byte. Minneskortet tilldelas rätt basadress med hjälp av en byglingsplugg som på 2055-kortet sitter i position 1D. Sker minnesexpansion med 8K byte och basadressen skall vara 40K byte anordnas avbrott på byglingspluggen enligt bild 29. Krävs ytterligare minnesexpansion klipps det andra kortets byglingsplugg enligt bild 30. Byglingspluggen kan även ersättas med en DIP-switch. När minneskortet placeras i expansionsutrymmets minnesdel skall komponentsidan vara åt höger. Platsen för kortet är valfri inom minnesutrymmets ram. Vid inkopplingstillfället bör datorsystemet vara spänningslöst. Efter inkopplingen

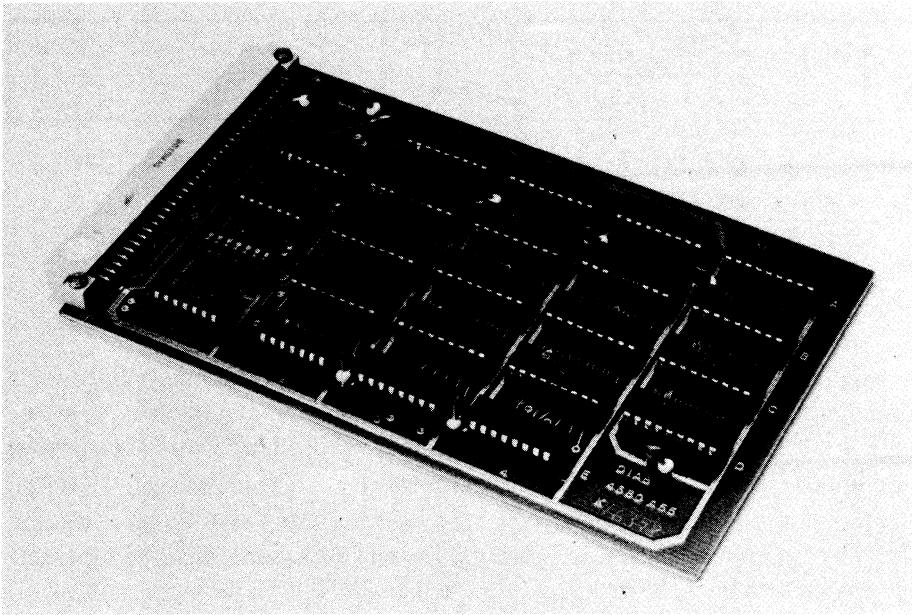


Bild 27. 8K byte av RAM-kort 2055.

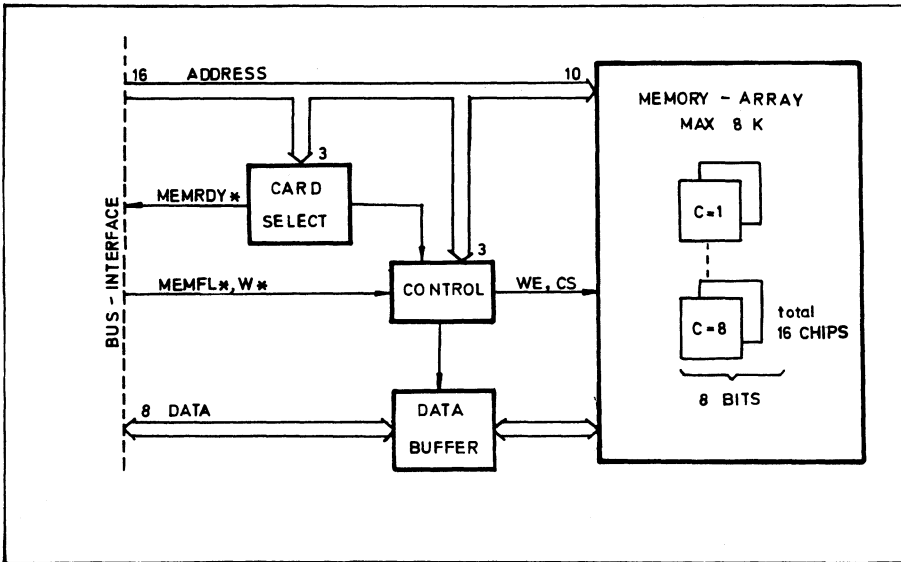
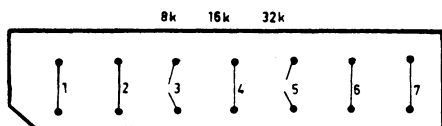
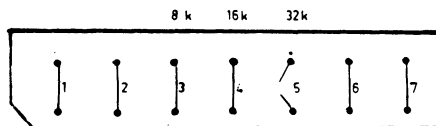


Bild 28. Blockschema för minneskort 2055.



Klipp bygling 3 och 5, ger
8k + 32k = 40k basadress.

Bild 29. Bygling av minneskort.



Klipp bygling 5, ger
32k basadress.

Bild 30. Bygling av minneskort.

av minneskortet bör man kontrollera att ABC80 har upptäckt inkopplingen och därmed utökningen av användarminnet. Slå på spänningen till ABC80-systemet. ABC:n tar nu reda på hur mycket minne som finns. Detta sker genom en rutin som först skriver i en minnescell och sedan läser i samma cell. Därefter jämförs det lästa med det inskrivna. Den basadress som ABC80 på detta sätt hittar lagras i minnescell 65053. Med hjälp av en PEEK-sats erhålls basadressen för användarminnet. Skriv

```
PRINT PEEK(65053)/4
```

ABC80 svarar då med att skriva 48, 40 eller 32 beroende på vilken basadressen är. Motsvarande resultat kan fås av

```
PRINT PEEK(65053)
```

I detta fallet svarar ABC80 med t ex 192. Detta värde anger antalet block om vardera 256 byte som basadressen ligger på. Genom att multiplicera 192 med 256 erhålls 49152 som är den decimala basadressen. Det är den minnesadressen som utgör "golvet" i RAM-arean. Basadressen blev alltså 48K byte. Är minnet expanderat med ett RAM-kort blir den nya basadressen 40K byte. Om man på nytt använder

```
PRINT PEEK(65053)
```

blir svaret denna gång 160 (160×256=40960). "Golvet" är alltså sänkt till 40K byte. Om man tillfälligt vill koppla bort externminnet för att återfå basadressen 48K byte kan man skriva

```
POKE 65053,192
```

Med detta kommando höjs "golvet" med 32 block, som är mellanskillnaden mellan 160 och 192 (32 block=8192 byte).

Med nedanstående program sker test av det inkopplade RAM-kortet. På bildskärmen kan man hela tiden följa undersökningen. I ett fönster visar

adressräknaren vilken minnesadress som undersöks och utskrift sker av eventuella fel. På rad 10 undersöks om basadressen är 48K byte. Är så inte fallet lyfts "golvet" så att programmet hamnar i internminnet, vilket är en förutsättning för att undersöka externminnet.

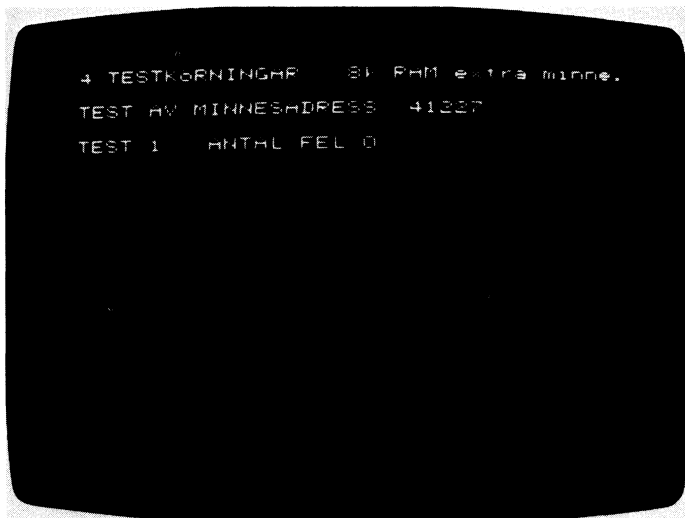


Bild 31. Inkopplingen av 8K byte RAM-kort kan undersökas med programmet "Testram".

TESTRAM

```

10 IF PEEK(65053) < 192 THEN POKE 65053, 192 : CHAIN "TESTRAM.BAC"
20 REM TESTRAM
30 REM TEST AV EXTERNT RAM TILL ABC 00
40 : CHR$(12);
50 : "8K ELLER 16K EXTRA MINNE ";
60 INPUT A
70 IF A=0% S=40960 : GOTO 100
80 IF A=16% S=32768 : GOTO 100
90 GOTO 40
100 IF PEEK(25000) < 255% 130
110 : CUR(9,0) "STARTA DATA DISC 00!!!": POKE 31912,164 : POKE 31937,164
120 IF PEEK(25000)=255% 120
130 : CHR$(12); "4 TESTKÖRNINGAR "A"K RAM EXTRA MINNE. " : : : "TEST AV MINNESADRESS "
140 FOR T%=0 TO 255 STEP 85
150 FOR I%=5 TO 49151
160 : CUR(2,21)65535+I%
170 POKE I%, T%
180 IF PEEK(I%) < T% F%=F%+1%
190 IF F%=0192% AND T%=0% : CHR$(12)CUR(9,0)"ÄR KORT FÖR MINNEEXP. "A"K MONTERAT" : : "PÅ RÄTT KORTPLATS ???": END
200 NEXT I%
210 : CUR(T%/85+5,0)"TEST"T%/85+1" ANTAL FEL"F% : F%=0 : : CUR(2,21)" : CHR$(7) : NEXT T%
220 : CUR(2,0)SPACE$(30)CUR(10,0)

```


Adressering av I/O-kort

I/O-bussens konstruktion gör det möjligt att styra inkopplingen av 64 olika I/O- eller anpassningskort med ABC80. Normalt är det inte aktuellt att använda så många kort. Direkt på ABC-bussen kan ett till två kort kopplas parallellt. Sker inkoppling på detta sätt utnyttjas spänningsmatningen från ABC80.

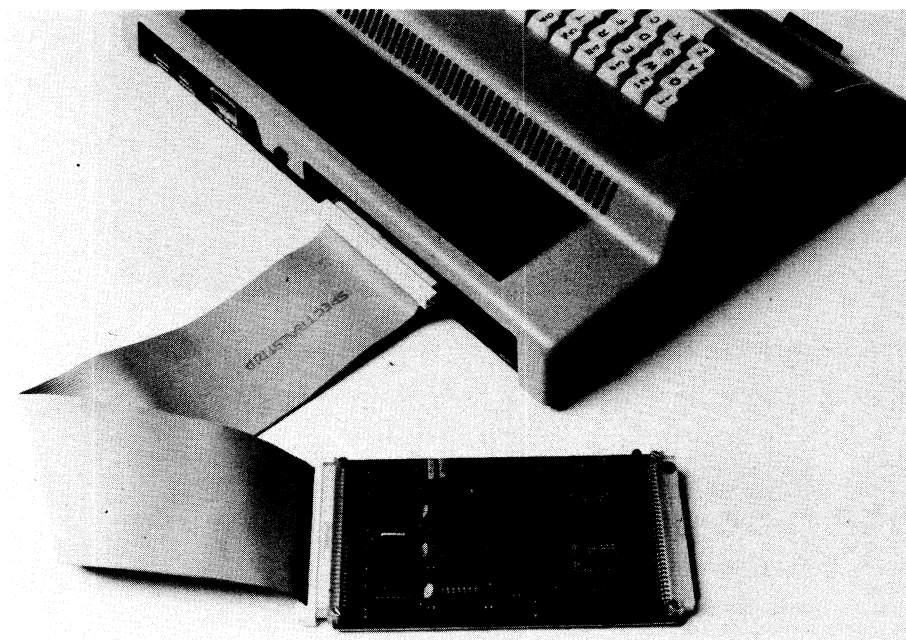


Bild 32. Inkoppling av ett I/O-kort kan ske direkt på bussen. I/O-kortet får då sin strömförsörjning från ABC80.

Kortens bestyckning och därmed strömförbrukning bestämmer därför antalet kort. Utnyttjas DataDisc 80 eller flexskiveminne ABC finns det tre I/O-platser i expansionsutrymmet och två i flexskiveminnet FD2. Valet av kort skall kunna ske med kommando eller som sats inskrivet i programmet. Kortvalet eller card select (CS), möjliggörs genom en koppling som sitter på varje kort.

När signalen CS (stift A 23) läggs ut på I/O-bussen tolkar alla kort på I/O-bussen innehållet på databussen. Om byglingsadressen på ett kort stämmer med innehållet på databussen är kortet utvalt.

CS-signalen fås från utport 1 och aktiveras med instruktionen

OUT 1,X

där X är ett tal mellan 0 och 63 beroende på vilken adress kortet tilldelats. Innan kortet placeras på plats, måste det ges en adress. På kortet finns därför en byglingsplugg (eller DIP-switch) där man klipper av önskade byglingar. Genom att addera bitvikterna för de avklippta byglingarna erhålls adressen.

Kvittering på att kortet är utvalt fås genom att en lysdiod på kortets framkant tänds. Kortet förblir sedan inkopplat ända tills nästa kortvalsinstruktion kommer. I och med att kortet är utvalt kan ABC80 skriva eller läsa data via kortet.

Frånkoppling eller reset av kortet sker med RST-signalen genom att skriva

PRINT INP(7)

eller genom att lägga ut en ny kortvalsinstruktion. Adressen på det sist utvalda kortet lagras i minnescell 65067, vilket kan kontrolleras med

PRINT PEEK(65067)

OBS. ABC80 skickar, via basicinterpretatorn, ut den sist angivna kortadressen varje gång INP eller OUT-instruktionen används. Detta inkluderar även OUT 6 (ljudgeneratorn) och V:24-snittet med OUT/INP 58. När man arbetar med program skrivna i assembler skickas alltså inte kortadressen automatiskt ut.

Anslutning av snabbskrivare

För att kunna ansluta en printer till ABC-bussen måste anpassning dem emellan ske. Detta sker genom ett anpassningskort (interface), som placeras mellan printern och ABC-busskontakten på ABC80. Interfacet är en I/O-tillämpning och konstruerat för det s k Centronics-snittet. Detta gör det möjligt att koppla in Centronics-printrar liksom printrar av andra fabrikat, som är försedda med Centronics-snittet. Bild 33 visar schemat för Luxors snabbskrivarinterface. Till vänster på schemat sker anslutning från ABC-bussen via en flatbandkabel. Interfacets utgång är till höger. För att printern skall fungera fordras inte bara en anpassning utan också ett drivprogram. Detta ligger i två PROM-kapslar (7611 och 7621). Då facet är kopplat direkt till ABC-bussen ligger även adressbussen kopplad till facet, vilket är ett avsteg från praxis för I/O-kort. Adressbyglingen sker i position 4A och har basadressen 30K byte. Eftersom facet är ett I/O-kort måste även kortval ske. I position 3C sitter därför byglingen för card select (CS), som normalt har den decimala adressen 60.

Som alternativ till inkoppling direkt på ABC-bussen kan interfacekortet placeras på valfri plats i flexskiveminne FD2. Bakkåpan på FD2 går dock inte att montera på plats eftersom detta saknar hål för I/O-kontakten. Om man använder DataDisc 80 eller flexskiveminne ABC tillsammans med ABC80 bör interfacekortet placeras på I/O-plats i flexskiveminnets expansionsutrymme. Eftersom det inte finns någon adressbuss framdragen till expansionsutrymmets I/O-del kommer adressbussingångarna att hänga i luften och därmed också drivprogrammet. Då printerrutinerna inte fungerar utan detta drivprogram är ett motsvarande placerat på PROM-kort 3033 som sitter längst till höger i expansionsutrymmet. Av bild 34 framgår placeringen av drivprogrammet.

Initiering av printer

Att ha uppfyllt villkoren för printerinkopplingen till datorsystemet är inte tillräckligt för en tillfredställande funktion. Printern måste också initieras, vilket betyder att radräkning och sidframmatning (formfeed) skall fungera. Om man har en text som är mer än ett A4-ark långt, dvs mer än 72 rader vill man normalt att utskriften avbryts 3—4 rader från slutet på det första pappret. Genom formfeed skall sedan utskriften fortsätta efter 3—4 radframmatningar på nästa papper. Justera pappret så att överkanten kommer mittför printerhuvudet. Skriv därefter som kommando eller instruktion

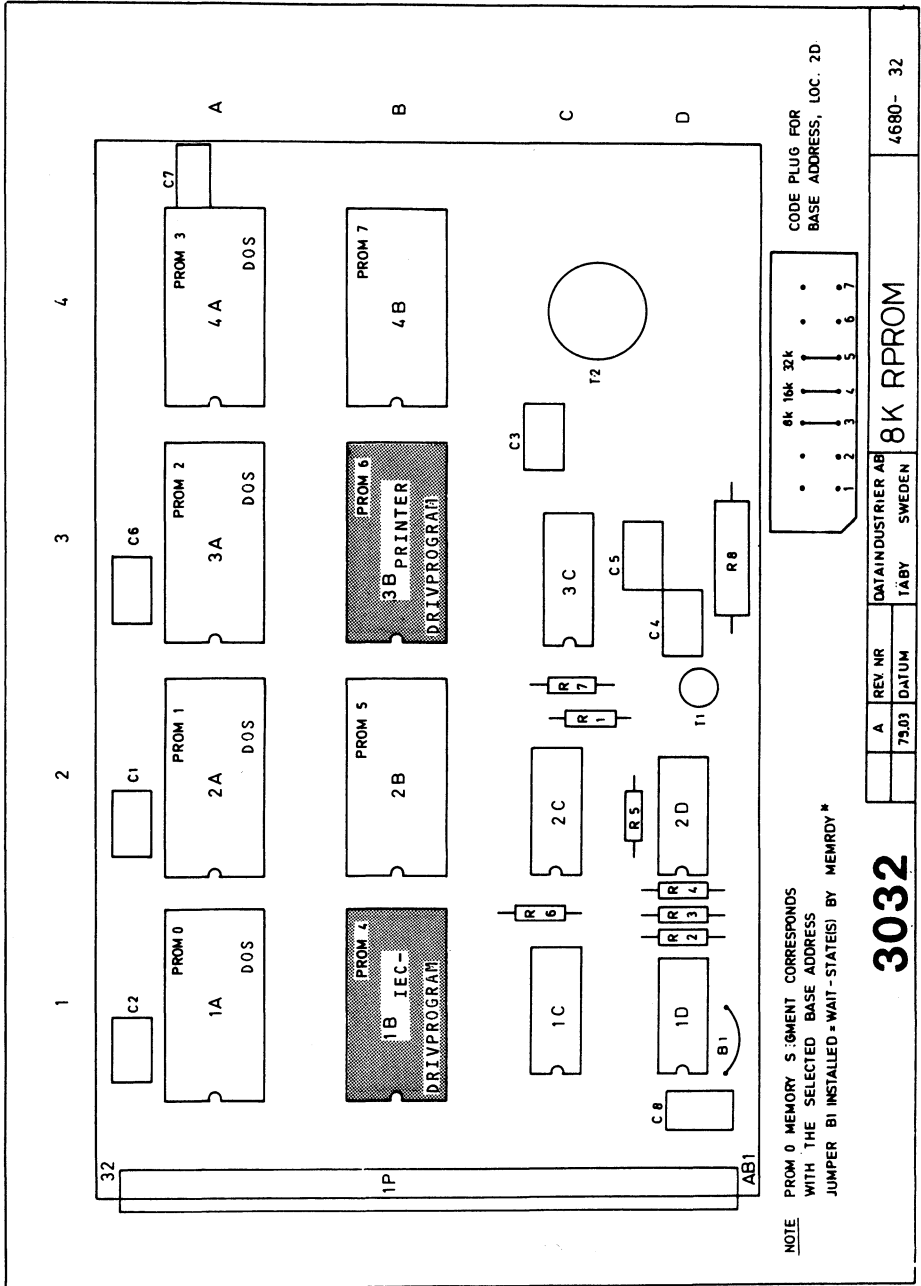


Bild 34. Placeringschema 8K byte PROM-kort 3032.

POKE 65037,200,0

Om pappret är kortare än 72 rader minskas 200 med motsvarande antal rader.
Om man vill se hur många rader som är skrivna kan man skriva

PRINT PEEK(65038)

Printern anropad med "PR:", exempelvis

OPEN "PR:" ASFILE 1

Om utskrift av programlistningen skall ske på printern skrivs

LIST PR:

Om man däremot skriver

LIST PR:A

görs ingen extra radframmatning eller formfeed. Utskriften i program kan också styras med ASCII-tecken enligt

PRINT #1 CHR □(10)=ny rad

PRINT #1 CHR □(12)=ny sida

PRINT #1 CHR □(13)=vagnretur

Exempelvis gör följande programrad två stycken radframmatningar

30 ;# CHR □(10,10)

Åtta reläutgångar

Med hjälp av I/O-kort 4007 kan lampor, kontakter, reläer, alarm etc styras elektriskt isolerat från datorsystemet. Varje utgång kan styras individuellt. Med ett kommando kan alla reläer brytas. På I/O-kortets I/O-sida finns en kontakt där samtliga reläutgångar är tillgängliga. Parallellt med dessa finns också åtta stycken TTL-utgångar. Genom bygling kan en extern spänningskälla användas till reläerna. Dessa matas normalt med en intern +12 volts spänning. Kontaktspänningen för reläerna är max 100 volt och kontaktströmmen max 0,5 Amp.

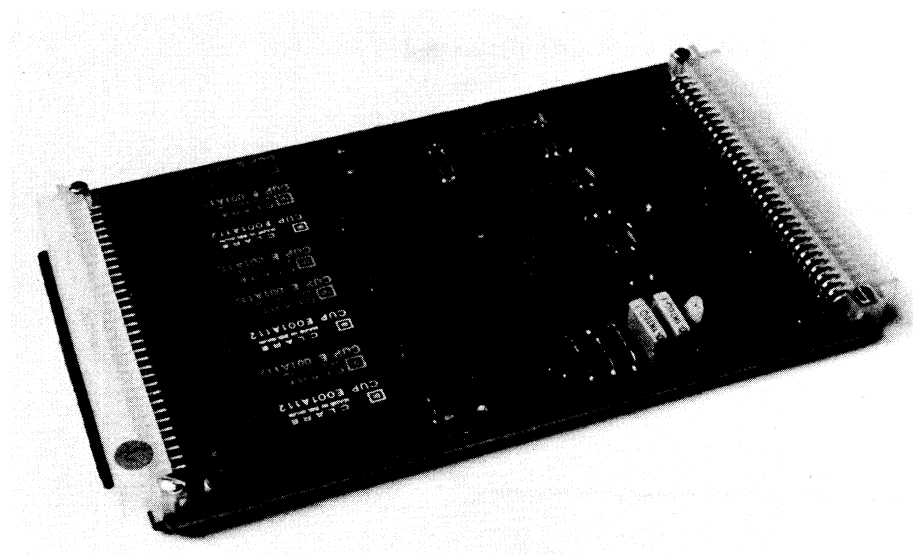


Bild 35. 8 utgångars reläkort 4007.

Inkoppling

För rätt kortval tilldelas I/O-kortet en kortvalsadress genom adressbygling. Byglingssluggen sitter på plats 2D. Om adressen skall vara t ex 7 bygglas kortet enligt bild 36. Slå av spänningen till datorsystemet och placera kortet i

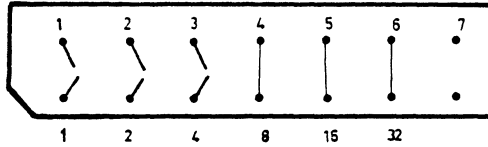


Bild 36. Bygling av 4007.

expansionsutrymmets I/O-del på valfri plats. Kontrollera inkopplingen genom att skriva

OUT 1,7

Om kortvalsadressen är 7 skall nu lysdioden på kortets framkant tändas. Skriv därefter

A=INP(7)

och lysdioden släcks vilket är lika med att kortet är frånkopplat.

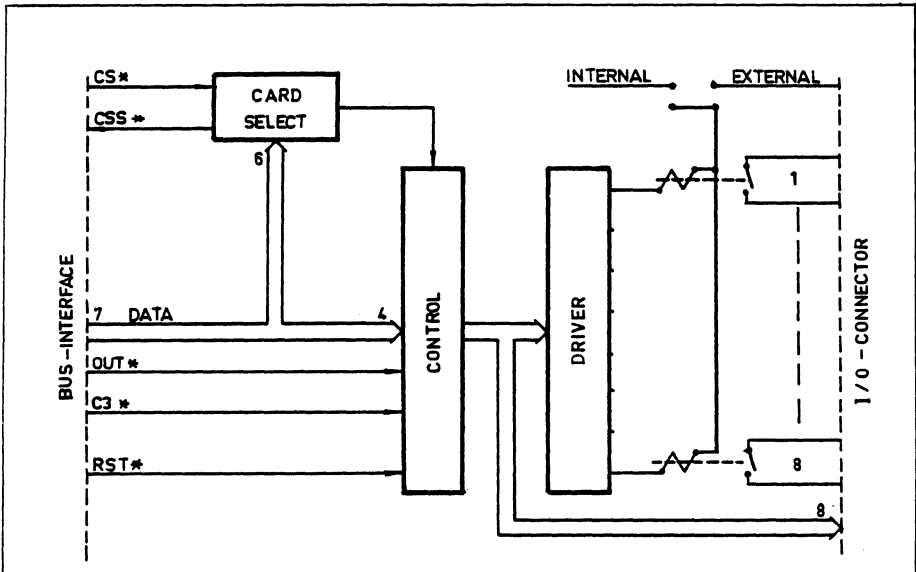


Bild 37. Blockschema reläkort 4007. C_{ss}=Card Select Special är en utsignal vilken kan användas som CS till eget I/O-kort.

I/O-kommandon för 4007

BASIC	SIGNAL SOM AKTIVERAS	ASSEMBLER	FUNKTION I BASIC
INP(7)	RST	IN 7	Nollställer alla I/O-kort. Släpper alla reläerna. 10 A=INP(7) Satsen bör finnas med i början på alla program.
OUT 1,A	CS	OUT 1	Väljer kort enligt A 20 OUT 1,7 Väljer ut kort med adress 7. När satsen utförs tänds lysdioden på det kort som adressat.
OUT 0,A	OUT	OUT 0	Reläkontakt A (0—7) bryts 30 OUT 0,1 Bryter reläkontakt 1.
OUT 0,A+128	OUT	OUT 0	Reläkontakt A (0—7) sluts. 40 OUT 0,2+128 Sluter reläkontakt 2
OUT 4,0	C3	OUT 4	Alla reläkontakter bryts 50 OUT 4,0

Laborationer med reläkort

För att bekanta sig med I/O-kort 4007:s möjligheter börjar man lämpligen med att göra en lysdiodramp bestående av åtta lysdioder. Dessa skall via kabel och kontakt anslutas till reläkortets I/O-kontakt. Lysdioderna kopplas lämpligen till TTL-utgångarna.

Exempel 1

I den här första övningen skall en lysdiod, som markerar respektive utgång, fås att blinka. Blinkfrekvensen skall dessutom kunna varieras. Lysdioden på kanal 1 skall blinka då programmet "Blink" exekveras. På programrad 20 är kortvalsadressen vald till 7. Reläkontakt 1 sluts på rad 30. Efter att ha genomlöpt en fördröjningsloop på rad 80 bryts reläkontakt 1 på rad 50. Samma fördröjning utnyttjas på nytt varefter reläet åter sluts.

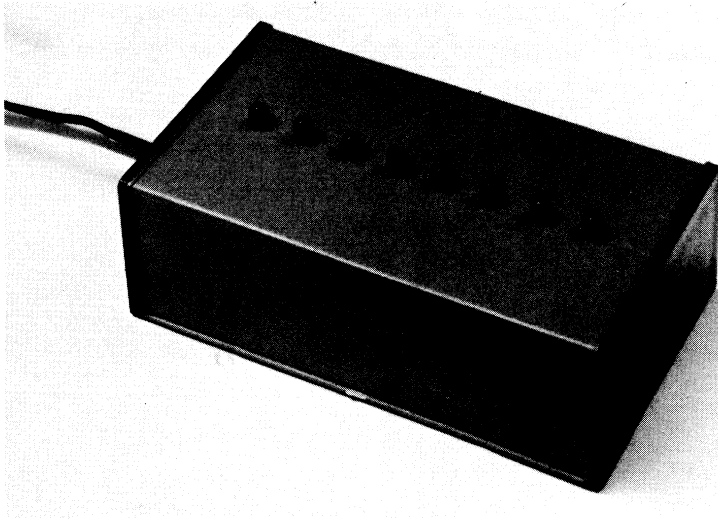


Bild 38. Som utorgan vid simulering med 4007 kan en lysdiodramp användas.

```
10 REM STVR 4007 ' BLINK '  
20 OUT 1,7 : REM CARDSELECT =7  
30 OUT 0,1+128  
40 GOSUB 80  
50 OUT 0,1  
60 GOSUB 80  
70 GOTO 30  
80 FOR I=1 TO 500 : NEXT I  
90 RETURN
```

Ändra på rad 30 så att något annat relä drar. Genom att dessutom ändra på rad 80 kan fördröjningen varieras.

Exempel 2

”Blink”-programmet är i nästa program utbyggt så att samtliga åtta utgångar skall kunna aktiveras i tur och ordning. På rad 40 kommer den utgång som skall aktiveras att bestämmas av innehållet i variabeln J. Dess värde går från 1—8 enligt loopen på raderna 30—80.

```
10 REM STVR 4007 ' RINGRÄKNARE '  
20 OUT 1,7 : REM CARDSELECT =7  
30 FOR J=1 TO 8  
40 OUT 0,J+128  
50 GOSUB 100  
60 OUT 0,J  
70 GOSUB 100  
80 NEXT J  
90 GOTO 30  
100 FOR I=1 TO 500 : NEXT I  
110 RETURN
```

Ändra, efter kontrollkörning, programmet så att "ringräknaren" räknar baklänges. Hur skall man göra för att endast varannan utgång skall aktiveras?

Exempel 3

Med programmet "Ramp" har det tidigare programmet kompletterats med satsen på rad 80. Denna rad gör att samtliga utgångar "noll-ställs".

```

10 REM STYR 4007 / RAMP /
20 OUT 1,7 : REM CARDSELECT =7
30 FOR J=1 TO 8
40 OUT 0, J+128
50 GOSUB 110
60 NEXT J
70 GOSUB 110
80 OUT 4, 0
90 GOSUB 110
100 GOTO 30
110 FOR I=1 TO 500 : NEXT I
120 RETURN
    
```

En del tillämpningskopplingar med reläkortet finns längre fram i boken.

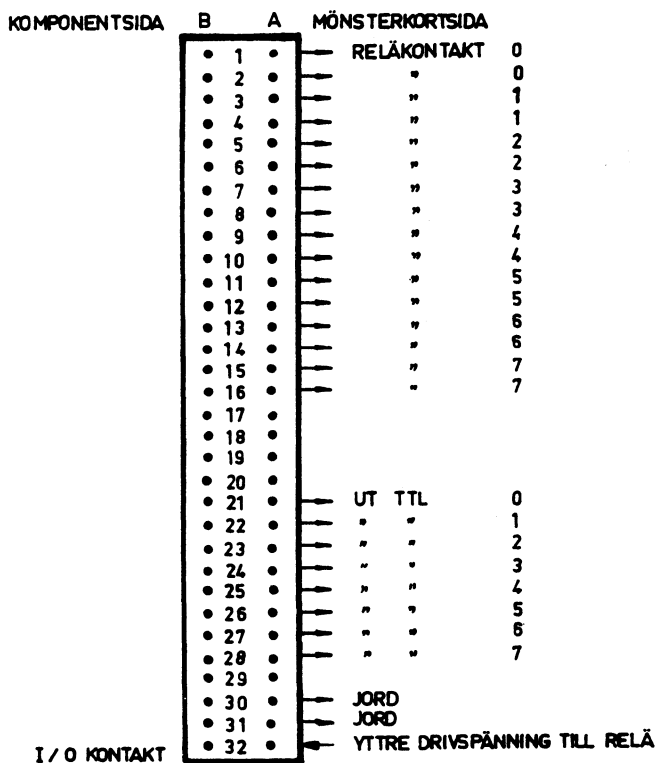


Bild 39. I/O-kontaktens stiftkonfiguration på 4007.

Analog-digitalomvandlare AD

I många sammanhang önskar man kunna ansluta en analog spänning till ett datorsystem. Av naturliga skäl är detta inte direkt möjligt. Den analoga spänningen måste först omvandlas till en digital motsvarighet. Detta sker i en AD(Analog-Digitalomvandlare eller Converter) eller är kanske beteckningen ADC mer korrekt. Det är inte alltid bara en analog spänning som skall anslutas.

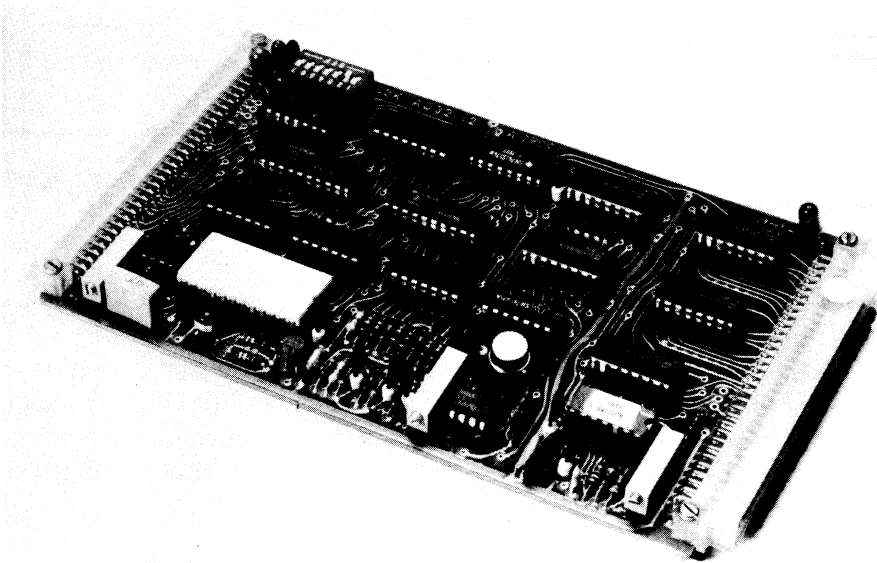


Bild 40. 32-ingångars AD.

Behovet är eventuellt att kunna ansluta många analoga spänningar. Den AD som i det följande beskrivs (4082) kan genomföra omvandling av informationen på 32 enkla ingångar eller 16 differentialingångar. Den analoga informationen till AD-ns ingångar kan erhållas från givarelement. Dessa omvandlar de fysikaliska storheter som skall mätas till elektriska signaler. Givarna kan vara

t ex Termoelement för mätning av temperatur

Fotoresistorer för mätning av ljus

Trådtöjningsgivare för mätning av kraft, moment eller läge

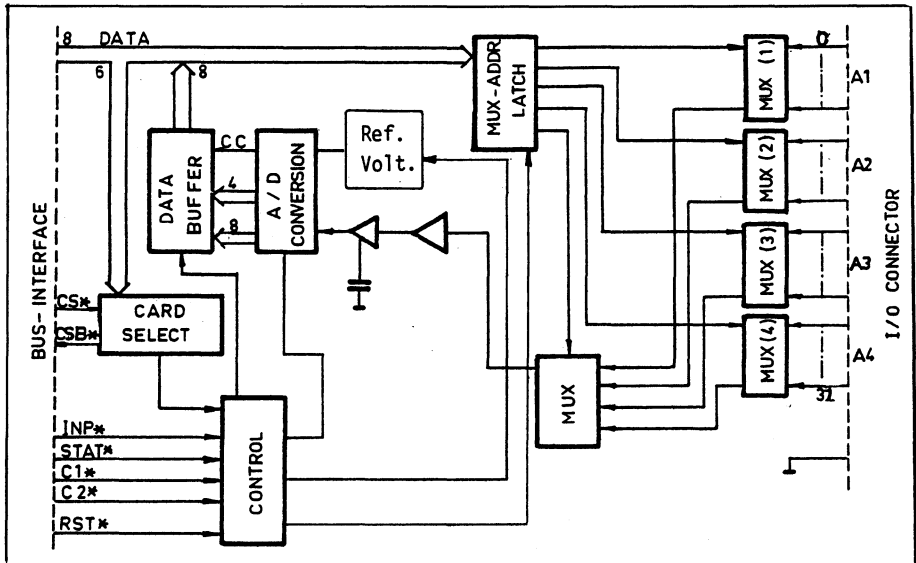


Bild 41. Blockschema AD 4082. CSB=Card Select Back Signal, används för backsignalering vid buffertval då I/O-utrymmet omfattar mer än 10 I/O-kort.

AD-n kan bl a användas för styrning av värmesystem, vägning, PCM-system, datalogger, transientrecorder och reglering av nivåer.

AD 4082 är baserad på omvandlingsmetoden "Successiv approximation". Upplösningen är 12 bitar, vilket medför att största inlästa värde blir 4095. Detta betyder att bitvärdet blir 1,2 mV vid 5 volts inspänning. Inspänningsområdet, som är programstyrt, kan vara 0—5 volt eller -5—5 volt. Max inström är 5 mA och omvandlingstiden 200 us.

Inkoppling av AD

För rätt kortval tilldelas AD-n en kortvalsadress. Om adressen skall vara 2 skall bygel 2 på byglingspluggen klippas (se bild 42). Slå av spänningen till datorsystemet och placera AD-kortet i expansionsutrymmets I/O-del på valfri plats. Kontrollera inkopplingen genom att skriva

OUT 1,2

Lysdioden på kortets framkant skall nu tändas. Skriv

A=INP(7)

och kortet blir fränkopplat, vilket markeras av att lysdioden släcks.

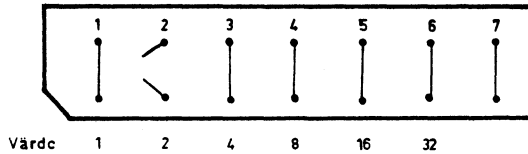
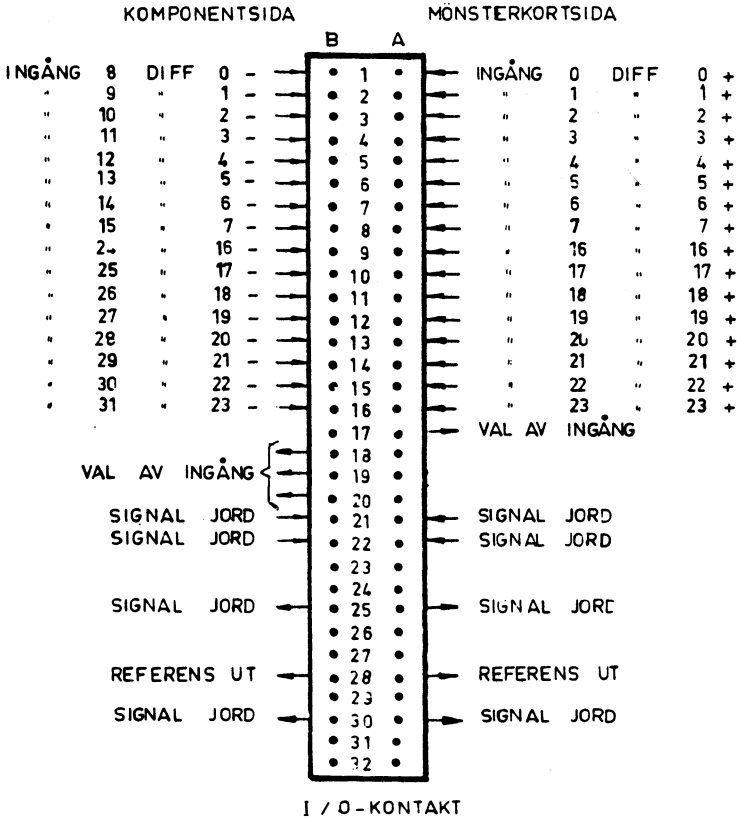


Bild 42. Bygling av AD 4082.



28 A/B REFERENS UT är testanslutning och får ej användas.

Bild 43. I/O-kontaktens stiftkonfiguration på AD 4082.

Till AD:s I/O-kontakt skall nu ett kontaktdon anslutas. För att de till kontakten anslutna mätpunkterna skall behandlas på rätt sätt måste kontakt-

donet byglas för val av separata eller differentiella ingångar. En av fyra nedanstående punkter skall utföras.

1. 32 ensamma ingångar.
Bygla 20B till 25A.
Bygla 19B till 25B.
2. 16 ensamma (0—15). 8 differential (16—23).
Bygla 20B till 25B.
Bygla 19B till 18B.
3. 8 differential (0—7). 16 ensamma (16—31).
Bygla 17A till 20B.
Bygla 19B till 25A.
4. 16 differential (0—7, 16—23).
Bygla 17A till 20B.
Bygla 18A till 19B.

I/O-kommandon för 4082

BASIC	SIGNAL SOM AKTIVERAS	ASSEMBLER	FUNKTION I BASIC
INP(7)	RST	IN 7	Nollställer alla I/O-kort. 10 A=INP(7) Satsen bör finnas med i början på alla program.
OUT 1,A	CS	OUT 1	Väljer kort enligt A. 20 OUT 1,2 Väljer kort med adress 2.
OUT 2,A	C1	OUT 2	A (0 till 31) anger adress för önskad analog ingång. Inspänningsområdet blir 0 till 5 V. 30 OUT 2,17 Väljer analog ingång nr 17.
OUT 2,A+32	C1	OUT 2	A (0 till 31) anger adress för önskad analog ingång. Inspänningsområde blir -5V till 5V. 40 OUT 2,32+3 Väljer analog ingång nr 3.
OUT 3,0	C2	OUT 3	Startar omvandling 50 OUT 3,0

INP(1)	STAT	IN 1	Inläsning av de fyra mest signifikanta bitarna.
INP(0)	INP	IN 0	Inläsning av de åtta minst signifikanta bitarna. 60 $R = INP(1) \times 256 + INP(0)$ Inläsning av samtliga 12 bitar. Lagrar värdet från analog ingång i variabel R.

Laborationer på AD

Innan AD-n appliceras i en systemtillämpning är det lämpligt med några enkla övningar i syfte att stifta en första bekantskap med AD-n.

Börja med att bygga AD-kortet för adress 2. Bygga därefter I/O-kontakten för 32 enkla ingångar. Anslut en spänningskälla 0—5 volt till ingång). Skriv in följande program.

```

10 REM AD
20 REM MÄTNING AV SPÄNNING 0 TILL 5 V
30 ; INP(7)
40 OUT 1, 2
50 ; CHR$(12)
60 ; CUR(10, 5); 'INSPÄNNING      VOLT'
70 OUT 2, 0, 3, 0
80 R=INP(1)*256+INP(0)
90 R=R/4095*5
100 R=INT(R*100+. 5)/100
110 ; CUR(10, 15); '      '; CUR(10, 15); R
120 GOTO 70
130 END

```

Då programmet körs skall utskrift ske på skärmen av den aktuella spänningen. Förklaring till vissa satser i programmet ovan:

- Rad 30 Kortet noll-ställs
- 40 Kortvalsadressen är 2.
- 70 2,0 väljer ingång 0 och spänningsområde 0—5 volt.
3,0 beordrar start av omvandling.
- 80 Hämtar mätvärdet.
- 90 Skalning.
- 100 Mätvärdesavrundning.

Med nästa program mäter AD-n spänningarna på de 20 första ingångarna. Programstrukturen är samma som i föregående program. På rad 80 väljs ingång beroende på innehållet i variabeln J. Denna variabel får sitt värde tilldelat från den loop som omfattar raderna 70—140.


```

10 REM AD
20 REM MÄTNING AV SPÄNNING 0 TILL 5 V
30 REM PÅ INGÅNGARNA 0-20
40 ; INP(7) : DIM R(20)
50 OUT 1,2
60 ; CHR$(12)"INGANG", "SPÄNNING VOLT"
70 FOR J=0 TO 20
80 OUT 2, J, 3, 0
90 R(J)=INP(1)*256+INP(0)
100 R(J)=R(J)/4095*5
110 R(J)=INT(R(J)*100+.5)/100
120 ; CUR(J+1, 0)SPACE$(40)
130 ; CUR(J+1, 0)J, R(J)
140 NEXT J
150 GOTO 70
160 END

```

Tillämpningskopplingar med AD återfinns på sidorna 84—90.

DAC

Ofta uppkommer behovet av att kunna styra en analog funktion med en dator. Eftersom datorn arbetar med digital information måste denna först göras om till en analog motsvarighet. För denna omvandling används en DAC (Digital Analog Converter).

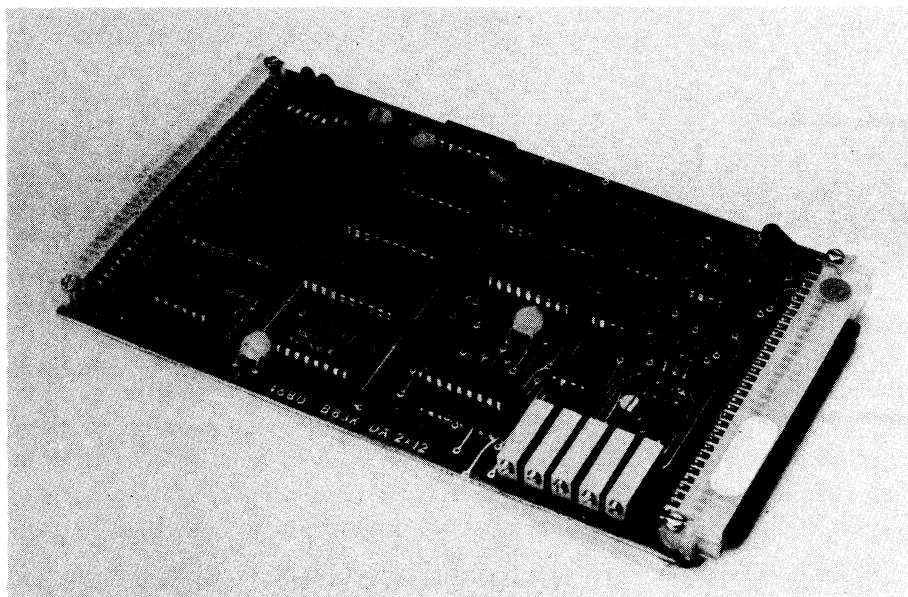


Bild 44. 2*12 bitars DAC 4083.

Ett exempel på användning är där DAC-n används för att styra en funktionsgenerator vilken i sin tur skall användas för att svepa en förstärkare. Med lämplig programbyggnad kan DAC-n fås att generera ett med tiden linjärt stigande spänningssvep. Genom att ansluta DAC-ns utgång till funktionsgeneratorns VCO-ingång (Voltage Controlled Oscillators) erhålls ett datorstyrt frekvenssvep.

Vanligt är att en DAC har flera kanaler för att kunna styra flera samtidigt analoga funktioner. Varje kanal har dessutom, som regel, två utgångar — en

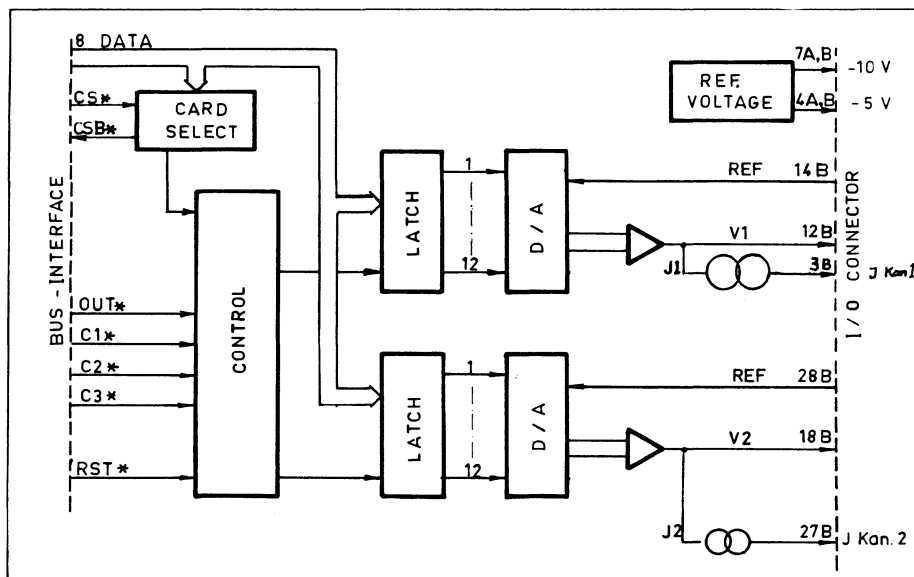


Bild 45a. Blockschema för DAC 4083.

ström utgång och en spänningsutgång. Den DAC som här skall beskrivas är Sattco:s 4083 som är $2 \cdot 12$ bitars DAC. Det I/O-kort som 4083 utgör kan placeras i expansionsutrymmet i DataDisc 80 eller FD2. Möjligheten finns också att via en flatkabel ansluta kortet direkt till busskontakten på ABC80.

På kortet finns två referensspänningar tillgängliga -5 och -10 volt. Dessutom kan yttre referensspänning inom intervallet ± 12 volt användas. Beroende på val av referensspänning kommer DAC-ns utsignal att variera enligt följande:

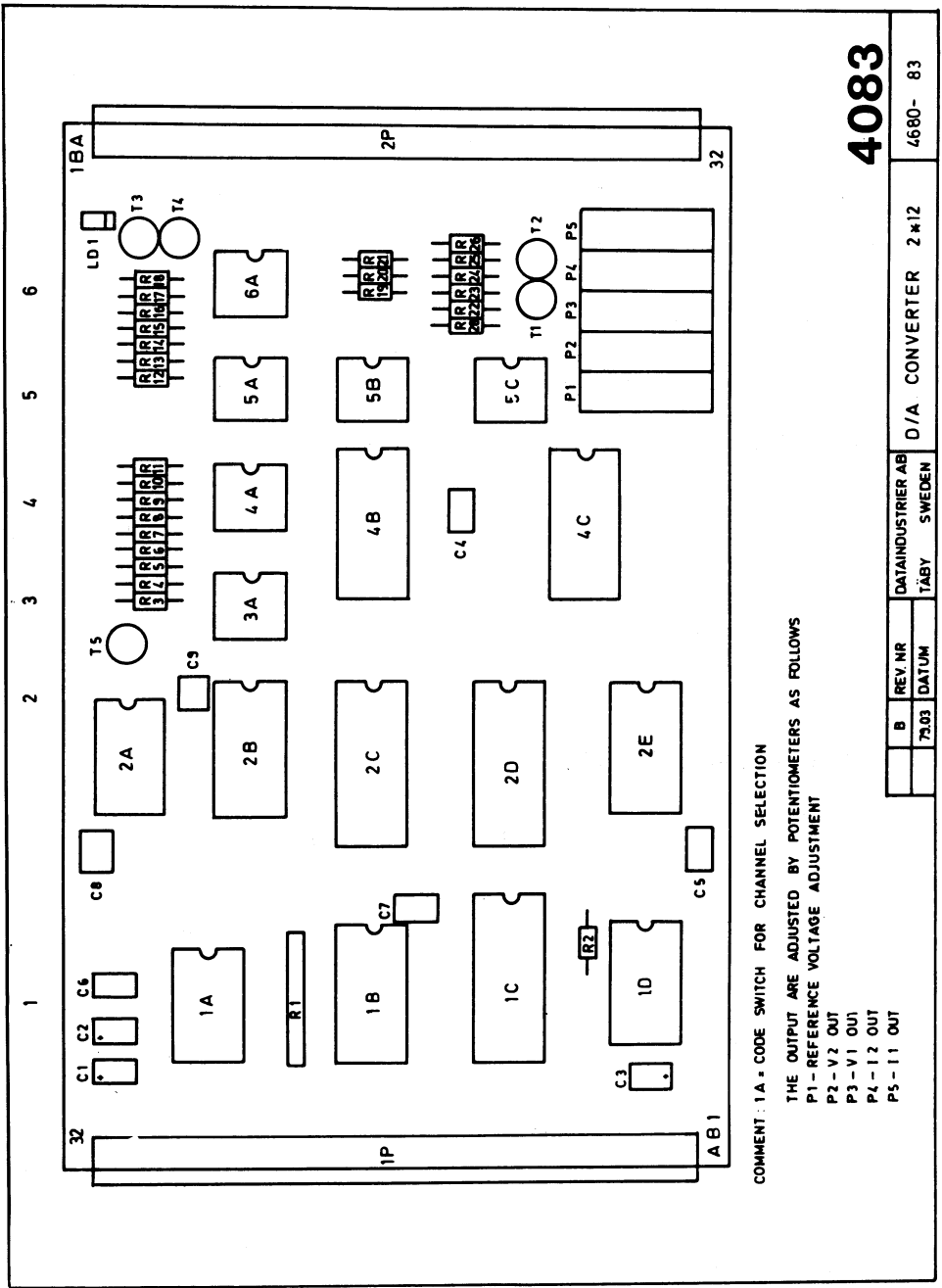
- 0—10 volt vid -10 volts referensspänning
- 0—5 volt vid -5 volts referensspänning

Respektive kanals ström utgång får belastas från 0—500 ohm. Utsignalen kommer då att kunna varieras 0—20 mA.

Bitvärdet för en 12 bits DAC blir referensspänningen/4096. Använder man 5 volts referensspänning blir upplösningen per bit= $1,2$ mV. Inställningstiden är cirka $5 \mu\text{s}$.

Innan kortet tas i bruk måste följande utföras:

1. Bestäm adress för kortet
2. Bestäm önskad referens genom bygling på korets I/O-sida (närmast lysdioden).



COMMENT : 1A - CODE SWITCH FOR CHANNEL SELECTION

THE OUTPUT ARE ADJUSTED BY POTENTIOMETERS AS FOLLOWS

P1 - REFERENCE VOLTAGE ADJUSTMENT

P2 - V2 OUT

P3 - V1 OUT

P4 - I2 OUT

P5 - I1 OUT

4083

4680- 83

D/A CONVERTER 2*12

DATAINDUSTRIER AB
TABY SWEDEN

B	REV.NR
79.03	DATUM

Bild 45b.

	OMRÅDE	BYGLA STIFT	UTSIGNAL STIFT
KANAL 1	0—10 V	7A—14B	12B
	0— 5 V	4A—14B	12B
	0—20 mA	4A—14B	3B
KANAL 2	0—10 V	7B—28B	18B
	0— 5 V	4B—28B	18B
	0—20 mA	4B—28B	27B

3. Anslut kortet till ABC-bussen.
4. Anslut yttre enhet till korets I/O-kontakt.
5. Kontrollera adresseringen till kortet genom att skriva

OUT 1,X

Om adresspluggen på kortet är byglad för adressen 3 skall lysdioden tändas om man skriver

OUT 1,3

Ges annat adressval skall lysdioden släckas.

I/O-kommandon för 4083

BASIC	SIGNAL SOM AKTIVERAS	ASSEMBLER	FUNKTION I BASIC
INP(7)	RST	IN 7	Nollställer alla I/O-kort. 10 A=INP(7) Satsen bör finnas med i början på alla program.
OUT 1,A	CS	OUT 1	Väljer kort enligt A. 20 OUT 1,3 Väljer kort med adress 3.
OUT 0,A%	OUT	OUT 0	Byte 2 av heltalsvariabeln A% skickas för omvandling till kanal 1 där den anger de minst signifikanta bitarna (7—0).
OUT 2,A%	C1	OUT 2	Byte 2 av heltalsvariabeln A% skickas för omvandling till kanal 1, där den anger de mest signifikanta bitarna (11—8).

			30 OUT 0,X,2,SWAP%(X) Skickar ut innehållet i variabel X för omvandling till kanal 1.
OUT 3,A%	C2	OUT 3	Samma som för OUT 0,A% men för kanal 2.
OUT 4,A%	C3	OUT 4	Samma som för OUT 2,A% men för kanal 2. 40 OUT 3,560,4,SWAP%(560) Skickar ut talet 560 för omvandling till kanal 2.

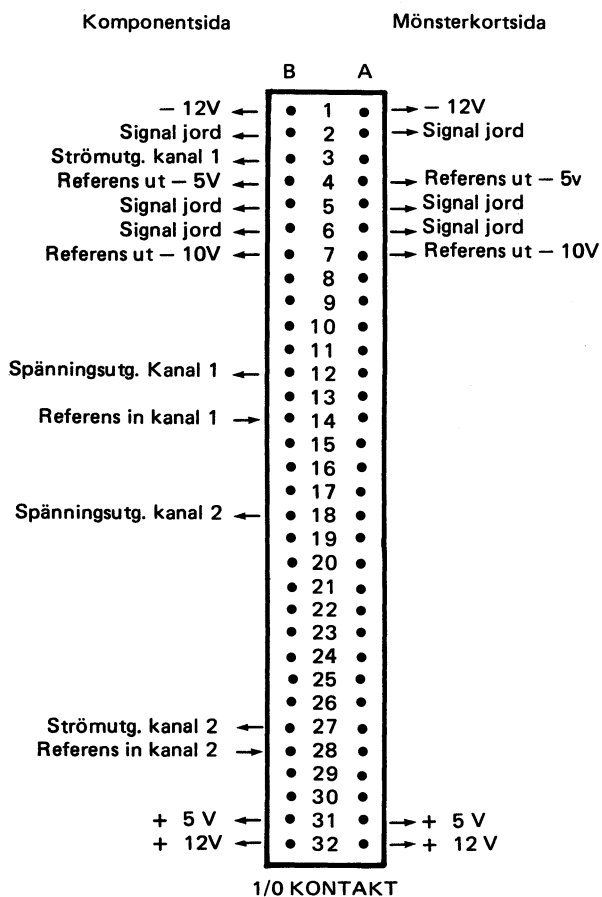


Bild 46. I/O-kontaktens stiftkonfiguration DAC 4083.

För att komma ut från kortet fordras tillgång till en 64-polig hon-kontakt. Bygla kontakten enligt bild 47 och ordna för anslutning av instrument. Byglingen gjord för 10 volts referensspänning på kanal 1 och 0—20 mA på kanal 2. I de två närmast följande programmen, som medger testning av kortfunktionerna, har kortvalet fått adressen 3 varför kortet bör vara byglat för samma adress. Anslut kort och en DMM till kanal 2. Skriv in och kör följande program.

```
10 A=INP(7)
20 OUT 1,3
30 PRINT "ANGE STRÖM I mA 0—20"
40 INPUT I
50 I%=I*4095/20
60 OUT 3,I%,4,SWAP%(I%)
70 GO TO 30
80 END
```

När programexekveringen hunnit fram till rad 40 frågas efter ett strömvärde mellan 0 och 20 mA. Skriv t ex 5 och iakttag visningen på instrumentet som nu bör vara lika.

För att prova spänningsutgången på kanal 1 kopplas DMM över till denna kanal och ställs i läge spänningsmätning 10 volt. Skriv därefter in och kör följande program.

```
10 A=INP(7)
20 OUT 1,3
30 PRINT "ANGE SPÄNNING I VOLT 0—10"
40 INPUT I
50 I%=I*4095/10
60 OUT 0,I%,2,SWAP%(I%)
70 GO TO 50
80 END
```

Uttrycket SWAP% i de två föregående programmen förenklar inställningen av DAC-ns utspänning. SWAP% innebär att man skiftar byte 1 med byte 2 i heltal. Eftersom databussen endast är en byte bred och heltal(%) innehåller två byte gäller att innehållet i byte 2 överförs vid en OUT-sats. För att de åtta minst signifikanta bitarna (LSB) av talet 555 ska skickas till port 0 och de 4 mest signifikanta bitarna (11—8) till port 2, dvs kanal 1, skriver man

```
60 OUT 0,555
70 OUT 2,SWAP%(555)
```

Att skriva på detta sätt innebär en ganska lång inställningstid för DAC-n. För

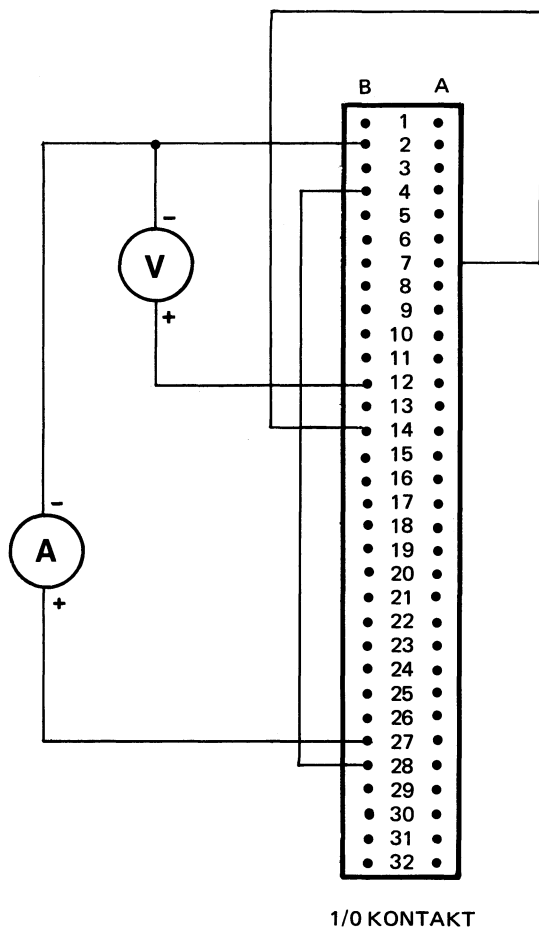


Bild 47. Mätuppkoppling.

att reducera denna till ca $250 \mu\text{s}$ skriver man istället

60 OUT 0,555,2,SWAP%(555)

Med nästa program skapar DAC-n en spänningsramp. Tiden för rampens tillväxt bestäms av loopen på raderna 50 och 100. Genom att ändra värdet 0.1 på rad 50 kan hastigheten ökas respektive minskas. På rad 70 sker skalning och på rad 80 skickas innehållet i variabeln S till kanal 1. Observera att kortvalsadressen på rad 60 har valts till 2. Kontrollera därför kortets adressbygling.


```

10 REM ** SPÄNNINGSRAMP **
20 REM ** DAC 4083 **
30 ; CHR$(12)
40 A=INP(7)
50 FOR R=0% TO 10% STEP .1
60 OUT 1,2
70 S=R*4095/10
80 OUT 0,S,2,SWAP%(S)
90 ; CUR(0,1)R : IF R=10 THEN ; CUR(0,1)SPACE$(5)CUR(0,0)R
100 NEXT R

```

För att få ner inställningstiden till ca 5 μ s kan någon av assemblerrutinerna nedan användas.

	ADRESS	OBJ CODE	MNEMONICS
FÖR KANAL 1	65408	123	LD A,E
	65409	211	OUTN,A
	65410	0	
	65411	122	LD A,D
	65412	211	OUTN,A
	65413	2	
	65414	201	RET
FÖR KANAL 2	65415	123	LD A,E
	65416	211	OUTN,A
	65417	3	
	65418	122	LD A,D
	65419	211	OUTN,A
	65420	4	
	65421	201	RET

Dessa två rutiner kan läggas i POKE-arean med början på adress 65408 genom att skriva

```

10 POKE 65408,123,211,0,122,211,2,201
20 POKE 65415,123,211,3,122,211,4,201

```

Anrop sker med CALL-funktionen genom att skriva

```

100 A=CALL(65408,555)+CALL(65415,X)

```

Värdet 555 skickas till kanal 1 medan innehållet i variabeln X skickas till kanal 2.

En naturlig utveckling är nu att låta DAC-n generera en svepspänning. Detta kan ske med nästa program som använder den tidigare beskrivna assembler-rutinen. Anslut oscilloskop eller XY-skrivare till utgången på kanal 1. Kör programmet och mät periodtiden. Ta därefter bort heltalsuttrycket (%) och mät på nytt periodtiden.

```

10 REM SVEP
20 REM GENERERING AV SVEPSPÄNNING
30 A=INP(7)
40 OUT 1,3
50 POKE 65415,123,211,0,122,211,2,201
60 INPUT I
70 FOR I%=0 TO 4095
80 A%=CALL(65415%,I%)
90 NEXT I%
100 GOTO 70

```

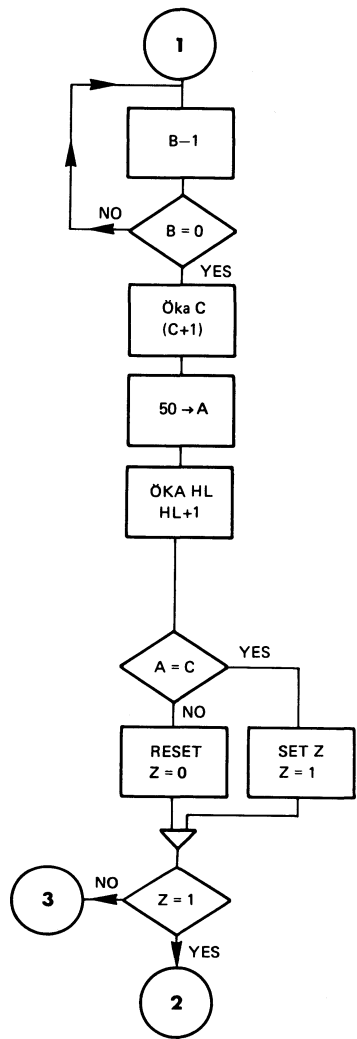
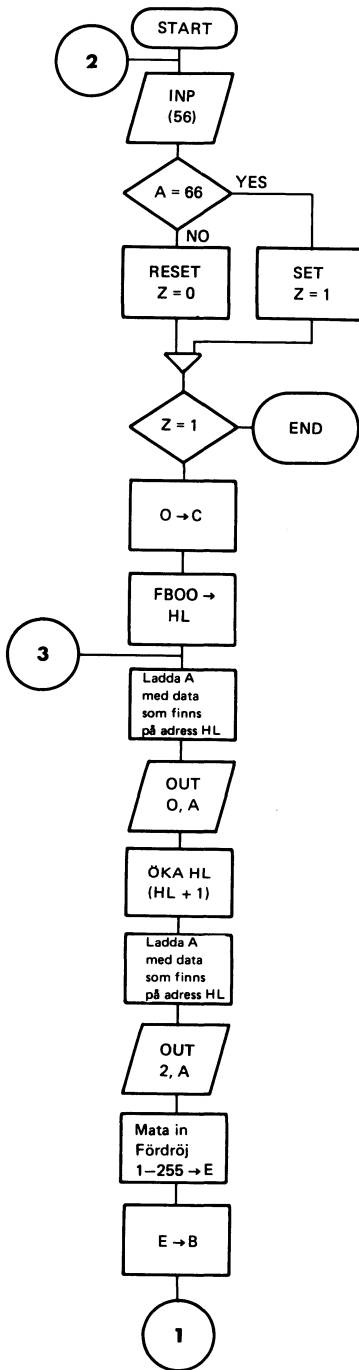
Som synes kan DAC-n användas för att generera en periodisk funktion. Med nästa program genereras en sinussignal. För att kunna arbeta med högre frekvenser måste en del av programmet skrivas i assembler. Programmet lägger upp en sinustabell bestående av 50 punkter med början på adress 64256. Assemblerrutinen avbrytas vid nedtryckning av tangent B och möjlighet ges att ändra fördröjningen. Utsignalens frekvens kommer att vara mycket stabil eftersom ABC80 styrs av en kristallklocka. En avbrottsstörning från realtidsklockan inträffar tyvärr var 20:e ms. Men eftersom avbrottet endast varar ca 10 μ s kan det oftast försummas.

```

5 REM *** SINUS ***
10 A=INP(7)
20 OUT 1,3
30 OUT 1,2
40 FOR I%=0 TO 100 STEP 2
50 S=SIN(I%*2*PI/100)*.5+.5
60 K%=S*4095
70 POKE 64256+I%,K%,SWAP%(K%)
80 NEXT I%
90 POKE 65408,219,56,254,66,200,14,0,33
100 POKE 65416,0,251,126,211,0,35,126
110 POKE 65423,211,2,67,16,254,12,62,50
120 POKE 65431,35,185,32,239,195,128,255
130 ; CHR$(12)'FÖRDRÖJNING (1-255)';
140 ; CLR(20,9)'FÖR OMSTART TRYCK PÅ 'B''CUR(0,10)
150 ONERRORGOTO 130 : INPUT F
160 A=CALL(65408,F) : GET B$
170 ; A : GOTO 130
180 END

```

Flödesschema för programmet "Sinus"



Maskinspråksrutinen för programmet "Sinus"

ADRESS	MNEMONICS	OBJ CODE	KOMMENTARER
65408	IN 38	219 56	Ta in värde från tangentbordet i ACC A
65410	CP 'B'	254 66	Kontrollera om tangent 'B' är nedtryckt
65412	RETZ	200	Om Z=0 PC L-(SP), PC H-(SP+I)
65413	LD C,0	14 0	Ladda ACC med 0
65415	LD HL,N	33, 0, 251	Ladda REG (HL) med (FB00) 64526
65418	LD A(HL)	126	Ladda ACC med data som finns på adress HL
65419	OUT 0	211 0	OUT 0, A
65421	INC HL	35	Öka HL
65422	LD A(HL)	126	Ladda ACC A med data som finns på adress HL
65423	OUT 2	211 2	OUT 2, A
65425	LD B,E	67	Ladda ACC B med innehåll i ACCE
65426	DJNZ	16 254	Minska ACC B och hoppa om B <>0 65425
65428	INC C	12	Öka REG C
65429	LD A,50	62 50	Ladda ACC A med 50
65431	INC HL	35	Öka HL
65432	CP C	185	Dra ifrån ACC A SET Z om A-C=0 annars RESET Z
65433	JRNZ	32 239	Hoppa om Z=0 65417
65435	JP	195 128 255	Hoppa till START 65408

Anslut ett oscilloskop till utgången på kanal 1 och mät upp frekvensen vid olika fördröjningar. Genom att ändra på rad 50 kan andra funktioner än sinus erhållas.

I många styr-, regler- och mättillämpningar krävs ett system som består av mer än ett I/O-kort. I följande systemtillämpning styrs DAC-ns byglingar av ett reläkort (4007). Denna koppling möjliggör programstyrd ändring av referensspänningen på båda kanalerna. Dessutom kan inkoppling av ström-alternativt spänningsutgångarna göras för kanal 1 och 2. Kopplingen är intressant vid laborationsarbete där man behöver kunna ändra förutsättningarna snabbt. Enligt programraderna 400 och 740 är kortvalsadressen 3 för reläkortet och 2 för DAC-n. När programmet startas ges följande skärmutskrift:

VILKEN UTGÅNG ÖNSKAS

1=ström	k1	max 20 mA
2=ström	k2	max 20 mA
3=spänning	k1	max 10 volt
4=spänning	k2	max 10 volt
5=spänning	k1	max 5 volt
6=spänning	k2	max 5 volt

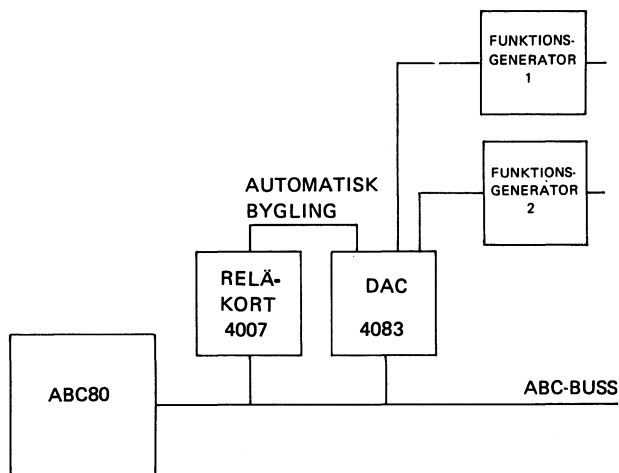


Bild 48. Systemuppkoppling.

Tryck på önskad siffertangent (utan RETURN). Enligt programrad 290 sker, beroende på tangentvalet, hopp till någon av programraderna 300, 350, 400, 450 eller 500 där bygling av DAC-n sker. Subrutinerna på programraderna 600 och 730 har samma struktur som programmen på sid 74. Kablaget för kopplingen mellan reläkortet och DAC-n framgår av bild 49.

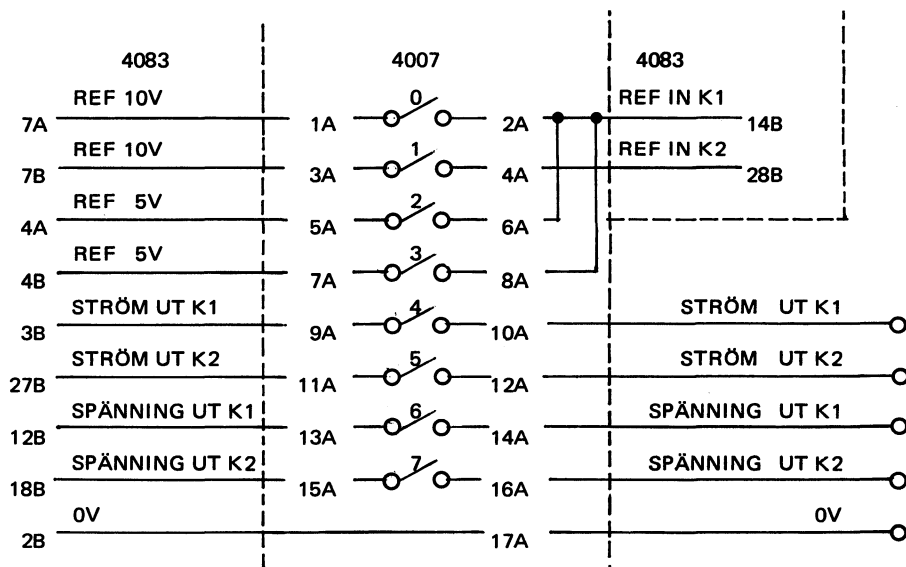


Bild 49. Kablagetschema.

```

10 REM *****
20 REM *       R E L Ä D A C           *
30 REM *                                           *
40 REM *                                           *
50 REM * STYRNING AV 4083 MED HJÄLP *
60 REM * AV RELÄKORT 4007.           *
70 REM *                                           *
80 REM * LARS KJELLGREN &           *
90 REM * ANDERS PALM                 *
100 REM *                                           *
110 REM * VASAGYMNASIET              *
120 REM * ARBOGA 1980-01-31          *
130 REM *                                           *
140 REM * 4007 HAR ADRESS 3          *
150 REM * 4083 HAR ADRESS 2          *
160 REM *                                           *
170 REM *****
180 ; CHR$(12)
190 A=INP(?) : REM NOLLSTÄLLER ALLA I/O-KORT
200 ; "VILKEN UTGÅNG ÖNSKAS"
210 ; "1=STRÖM K1 MAX 20MA"
220 ; "2=STRÖM K2 MAX 20MA"
230 ; "3=SPÄNNING K1 MAX 10V"
240 ; "4=SPÄNNING K2 MAX 10V"
250 ; "5=SPÄNNING K1 MAX 5V"
260 ; "6=SPÄNNING K2 MAX 5V"
270 GET K$
280 IF K$<"1" OR K$>"6" THEN 270
290 ON VAL(K$) GOTO 300,350,400,450,500,550
300 OUT 1,3 : REM VÄLJER IN RELÄKORT 4007
310 OUT 0,2+128 : OUT 0,4+128 : REM SLUTER RELÄ 2 & 4
320 B=0 : C=2
330 GOSUB 730
340 GOTO 180
350 OUT 1,3 : REM VÄLJER IN RELÄKORT 4007
360 OUT 0,3+128 : OUT 0,5+128 : REM SLUTER RELÄ 3 & 5
370 B=3 : C=4
380 GOSUB 730
390 GOTO 180
400 OUT 1,3 : REM VÄLJER IN RELÄKORT 4007
410 OUT 0,0+128 : OUT 0,6+128 : REM SLUTER RELÄ 0 & 6
420 B=0 : C=2 : R=10
430 GOSUB 600
440 GOTO 180
450 OUT 1,3 : REM VÄLJER IN RELÄKORT 4007
460 OUT 0,1+128 : OUT 0,7+128 : REM SLUTER RELÄ 1 & 7
470 B=3 : C=4 : R=10
480 GOSUB 600
490 GOTO 180
500 OUT 1,3 : REM VÄLJER IN RELÄKORT 4007
510 OUT 0,2+128 : OUT 0,6+128 : REM SLUTER RELÄ 2 & 6
520 B=0 : C=2 : R=5
530 GOSUB 600
540 GOTO 180
550 OUT 1,3
560 OUT 0,3+128 : OUT 0,7+128
570 B=3 : C=4 : R=5
580 GOSUB 600
590 GOTO 180
600 ; CHR$(12)
610 OUT 1,2 : REM VÄLJER IN D/A-KORTET 4083
620 ; "0=0MSTART, S=STOPP " : ;
630 ; "ANGE ÖNSKAD UTSPÄNNING I V (0-"R")"
640 ONERRORGOTO 640
650 INPUT U$
660 IF U$="S" OR U$="s" THEN 670 ELSE 680
670 ; CHR$(12) : CHAIN ""
680 IF U$="0" OR U$="o" THEN RETURN
690 U=VAL(U$)
700 U=U*4095/R : REM BESTÄMMER UPPLÖSNINGEN

```

```

710 OUT B, U, C, SWAP%<U> : REM VERKSTÄLLER
720 GOTO 640
730 : CHR$(12)
740 OUT 1, 2 : REM VÄLJER IN D/A-KORTET 4083
750 : "O=OMSTART, S=STOPP" : ;
760 : "ANGE ÖNSKAD UTSTRÖM I MA <0-20>"
770 ONERRORGOTO 770
780 INPUT I$
790 IF I$="S" OR I$="S" THEN 800 ELSE 810
800 : CHR$(12) : CHAIN ""
810 IF I$="0" OR I$="0" THEN RETURN
820 I=VAL<I$>
830 I=I*4095/20 : REM BESTÄMMER UPPLÖSNING
840 OUT B, I, C, SWAP%<I> : REM VERKSTÄLLER
850 GOTO 780

```

ABC80 som funktionsgenerator

En funktionsgenerator, visserligen med något begränsade funktioner, är vad ABC80 blir med nästa systemprogram. Kopplingen som används är samma som i föregående exempel. Kablaget för kopplingen, som består av en DAC och ett reläkort, framgår av bild 49. När programmet exekveras väljs från meny kanal och toppspänning. Därefter ges möjlighet att välja vågform som kan vara triangelvåg, tringelpuls, svepspänning, enkelsvep eller sinusvåg. Alstrandet av en del av dessa funktioner sker i programsekvenser som känns igen från sidorna 76—77.

```

10 REM ** KURVFORMER 4083+4007 **
20 : CHR$(12)
30 Q=INP(7)
40 : CUR(3,0)"VÄLJ KANAL OCH TOPPSPÄNNING"
50 : CUR(5,0)"1"CUR(5,5)"MAX 10 VOLT KANAL 1"
60 : CUR(6,0)"2"CUR(6,5)"MAX 10 VOLT KANAL 2"
70 GET A$
80 IF A$<"1" OR A$>"4" THEN 70
90 : CHR$(12)
100 ON VAL(A$) GOTO 110,130
110 : CUR(0,0)"MAX 10 VOLT KANAL 1"
120 GOTO 150
130 : CUR(0,0)"MAX 10 VOLT KANAL 2"
140 GOTO 150
150 REM
160 : CUR(3,0)"VÄLJ SIGNAL"
170 : CUR(5,0)"1"CUR(5,5)"TRIANGELVÅG"
180 : CUR(6,0)"2"CUR(6,5)"TRIANGELPULS"
190 : CUR(7,0)"3"CUR(7,5)"SVEPSPÄNNING"
200 : CUR(8,0)"4"CUR(8,5)"ENKELSVEP"
210 : CUR(9,0)"5"CUR(9,5)"SINUSVÅG"
220 GET B$ : IF B$="5" THEN 1050
230 IF B$<"1" OR B$>"5" THEN 220
240 L=INP(7)
250 ON VAL(A$) GOTO 260,270
260 E=0 : F=6 : G=0 : D=2 : GOTO 280
270 E=1 : F=7 : G=3 : D=4 : GOTO 280
280 : CHR$(12)
290 : "ANGE TOPPSPÄNNINGEN MAX<10 VOLT>"
300 ONERRORGOTO 280
310 INPUT A
320 IF A<10 THEN 280

```

```

330 IF A<0 THEN 280
340 ; CUR(7,1)*"ANGE STIGTIDEN (1-10)"
350 ONERRORGOTO 340
360 INPUT C
370 U1=0 : U2=A : S1=1/C
380 OUT 1,3
390 OUT 0,E+128 : OUT 0,F+128
400 ON VAL(B*) GOTO 420,690,830,950
410 REM ***** TRIANGEL VAG *****
420 REM A=INP(7)
430 FOR S=U1 TO U2 STEP S1
440 ; CUR(18,14)*"För STOPP TRYCK PÅ 'S'"
450 OUT 1,2
460 R=S*4095/10 : P=FIX(S)
470 OUT G,R,D,SWAP%(R)
480 ; CUR(10,17)P : IF S=U2 THEN ; CUR(10,14)SPACE*(9)CUR(10,16)
490 IF INP(56)=211 OR INP(56)=243 THEN ; CUR(18,14)SPACE*(23) : GOSUB 590 ELSE 500
500 NEXT S
510 FOR S=U2 TO U1 STEP -S1
520 R=S*4095/10 : P=FIX(S)
530 OUT G,R,D,SWAP%(R)
540 ; CUR(10,17)P : IF S=U2 THEN ; CUR(10,14)SPACE*(9)CUR(10,16)
550 IF INP(56)=211 OR INP(56)=243 THEN ; CUR(18,14)SPACE*(23) : GOSUB 590 ELSE 560
560 NEXT S
570 GOTO 430
580 REM ** SUBROUTIN För OMSTART/NYA VÄRDEN **
590 IF INP(56)=207 OR INP(56)=239 THEN ; CUR(19,13)SPACE*(67) : RETURN
600 ; CUR(19,13)*"För OMSTART TRYCK PÅ 'O'"
610 ; CUR(20,13)*"För NYA VÄRDEN TRYCK PÅ 'N'"
620 ; CUR(21,13)*"För AVSLUT TRYCK PÅ 'A'"
630 IF INP(56)=193 OR INP(56)=225 THEN ; CHR*(12) : CHAIN ""
640 IF INP(56)=206 OR INP(56)=238 THEN 660
650 GOTO 580
660 GET F$
670 GOTO 20
680 RETURN
690 REM ***** TRIANGEL PULS *****
700 FOR S=U1 TO U2 STEP S1
710 OUT 1,2
720 R=S*4095/10 : P=FIX(S)
730 OUT G,R,D,SWAP%(R)
740 ; CUR(10,17)P : IF S=U2 THEN ; CUR(10,14)SPACE*(9)CUR(10,16)
750 NEXT S
760 FOR S=U2 TO U1 STEP -S1
770 R=S*4095/10 : P=FIX(S)
780 OUT G,R,D,SWAP%(R)
790 ; CUR(10,17)P : IF S=U2 THEN ; CUR(10,17)SPACE*(9)CUR(10,17)
800 NEXT S
810 GOSUB 590 : GOTO 690
820 REM ***** SVEPSÄNNING *****
830 ; CUR(18,14)*"För STOPP TRYCK PÅ <S>"
840 OUT 1,3
850 OUT 0,E+128 : OUT 0,F+128
860 FOR S=U1 TO U2 STEP S1
870 IF INP(56)=211 OR INP(56)=243 THEN ; CUR(18,14)SPACE*(23) : GOSUB 590
880 OUT 1,2
890 R=S*4095/10
900 OUT G,R,D,SWAP%(R)
910 ; CUR(10,16)S : IF S=U2 THEN ; CUR(10,16)SPACE*(5)
920 NEXT S
930 GOTO 860
940 REM ***** ENKELSYEP *****
950 FOR S=U1 TO U2 STEP S1
960 OUT 1,2
970 R=S*4095/10
980 OUT G,R,D,SWAP%(R)
990 ; CUR(10,16)S : IF S=U2 THEN ; CUR(10,16)SPACE*(5)
1000 NEXT S
1010 R=0 : OUT G,R,D,SWAP%(R)
1020 ; CUR(10,16)SPACE*(5)CUR(10,17)"0"

```



```

1030 GOSUB 590
1040 GOTO 950
1050 REM ***** SINUSVAG *****
1060 OUT 1,3 : OUT 0,0+128 : OUT 0,6+128
1070 OUT 1,2
1080 FOR I%=0 TO 100 STEP 2
1090 S=SIN(I%*2*PI/100)*.5+.5
1100 K%=S*4095
1110 POKE 64256%+I%,K%:SNAP%(K%)
1120 NEXT I%
1130 POKE 65408,219,56,254,66,200,14,0,33
1140 POKE 65416,0,251,126,211,0,35,126
1150 POKE 65423,211,2,67,16,254,12,62,50
1160 POKE 65431,35,185,32,239,195,128,255
1170 ; CHR$(12)'FÖRDRÖJNING (1-255)';
1180 ONERRORGOTO 1170 : INPUT F
1190 ; CUR(20,5)'FÖR NY FREKVENNS TRYCK PÅ 'B'*CUR(0,10)
1200 REM ONERRORGOTO 1160 : INPUT F
1210 A=CALL(65408,F) : GET B$
1220 ; CUR(20,5)SPACE$(40) : GOSUB 580 : GOTO 1170
1230 END

```

System för komponentprovning

Inom industrin används i ökad utsträckning olika typer av automatiska mätsystem. Utrustningar för automatisk komponentprovning är ett exempel på ett sådant mätsystem. Kopplingen på bild 50 visar principen för mätning av resistorer. DAC-n lägger ut en spänning till en spänningsdelarkoppling där en

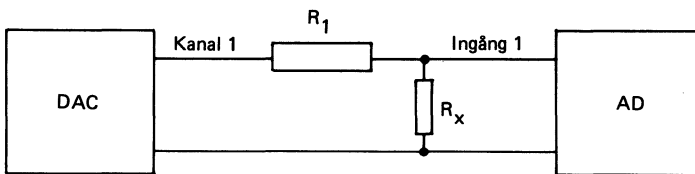


Bild 50. Principen för resistansmätning med DAC och AD.

AD (Analog-Digitalomvandlare) mäter spänningsfallet över den okända resistorn R_x . Genom att ändra storleken på referensresistorn R_1 kan resistansmätområdet ändras uppåt respektive nedåt. Om man skall mäta ett antal resistorer av viss storlek kanske man vill rensa ut de resistorer som inte ligger inom en viss toleransgräns. Programmet kan då kompletteras så att akustisk larmsignal ges om det aktuella mätvärdet ligger över eller under toleransgränserna. Byggingen av korten sker enligt bild 52.

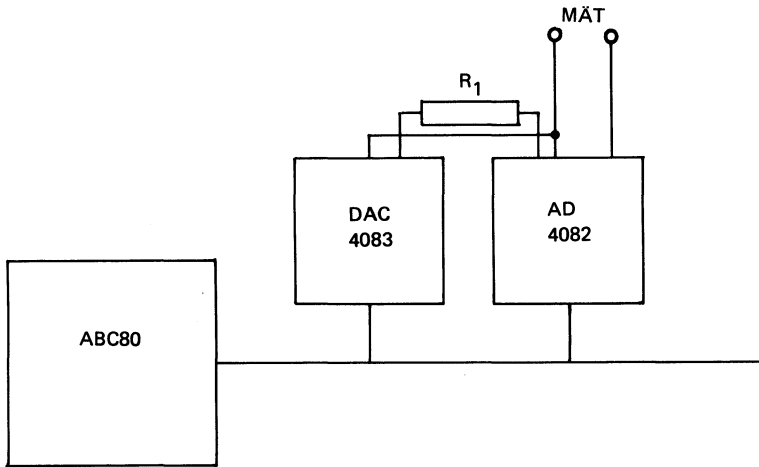


Bild 51. Systemschema.

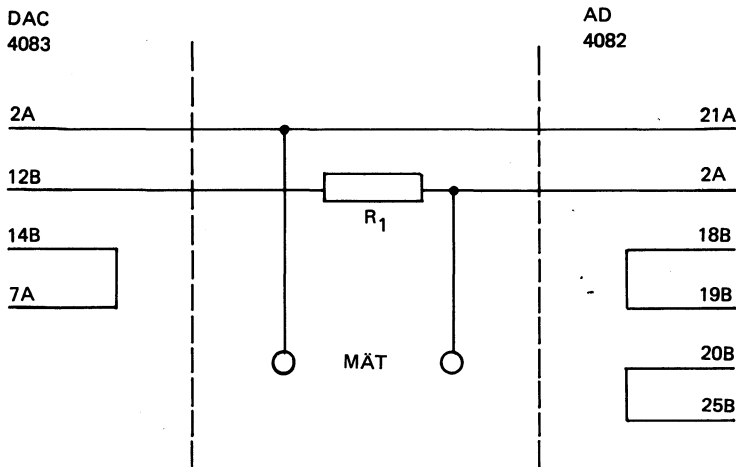


Bild 52. Kablagescema.

```

10 ; CHR#(12)
20 REM RESISTANSMÄTNING MED DAC OCH AD
30 REM AD BYGLAS 18B-19B, 20B-25B
40 REM DAC BYGLAS 7A-14B
50 REM KOPPLA EN REFERENSRESISTOR
60 REM MELLAN AD 2A OCH DAC 12B
70 REM KOPPLA AD 21A TILL DAC 2B
80 REM RESISTANSEN MÄTS MELLAN
90 REM AD 2A OCH AD 21A
100 R1=4700 ; REM R1= REF RESISTANSEN
110 REM SPÄNNINGSINSTÄLLNING

```

```

120 FOR I=0 TO 10 STEP .5
130 IX=I*4095/10
140 OUT 1,3,0,IX,2,SWAPX<IX>
150 GOSUB 180
160 IF U>=2.5 THEN 230
170 NEXT I : I=I-.5 : GOTO 230
180 REM MÄTER SPÄNNINGEN
190 OUT 1,2,2,1,3,0
200 U=INP<1>*256+INP<0>
210 U=U*5/4095
220 RETURN
230 REM BERÄKNA RESISTANSEN
240 U1=I-U : I1=U1/R1
250 IF U1<.1 : CUR<5,5>"ÖPPEN INGÅNG
260 IF U<.1 : CUR<5,5>"FÖR LÅG RESISTANS
270 : CUR<5,19>SPACE$(12)
280 : CUR<5,5>"RESISTANSEN ="U/I1" OHM"
290 GOTO 120

```

Automatisk mätvärdesinsamling

Följande systemtillämpning utgör exempel på ett mät- och styrsystem. Med ett 32-kanalers AD-kort (4082) mäts temperaturen från åtta mätställen. Som temperaturgivare används termistorer (ca 100 ohm vid 25°C). När programmet startas skall önskade gränsvärden för de olika mätpunkterna anges. För att underlätta orienteringen av skärmutskriften tilldelas de olika mätpunkterna namn som ligger i datasatser på programraderna 690—710. Till ABC-bussen är, utöver AD-n, även ett åtta-kanalers reläkort inkopplat. Någon av dess utgångar skall aktiveras om temperaturen i en mätpunkt är högre än mätpunktens gränsvärde. Vill man "vända" på denna funktion så att aktivering av reläkortets utgångar sker när temperaturen är lägre än gränsvärdet byter man tecken >: på rad 750 mot <.

Om fel uppstår på någon termistor eller på ledningen till denna ges akustisk larmsignal. Samtidigt erhålls på skärmen utskrift av ett felmeddelande som informerar om det är avbrott eller kortslutning. Om felet försvinner eller avhjälpas fortsätter övervakningen. På skärmen kommer en markör(>) att finnas kvar för att markera vid vilket mätställe felet har varit. Samtidigt som ett fel uppstår aktiveras också motorstyrningsutgången på ABC80 varför denna kan användas för att aktivera en larmanläggning. Enligt schemat, bild 55, kopplas en extern referensspänning till kanal 31. För att erhålla en acceptabel mätnoggrannhet mäter AD-n referensspänningen i varje mätloop. Är denna för låg skrivs detta ut på skärmen.

Bygla AD-n för 32 enkla ingångar dvs 20B till 25A och 19B till 25B. Enligt satsen på rad 280 är kortvalsadressen 2 för AD-n. För reläkortet är kortvalsadressen 7 enligt rad 740. Tillståndet på reläkortets utgångar kan indikeras av lysdioder anslutna till kortets TTL-utgångar.

MÄTPUNKT	TEMP	GRÄNS	STATUS
1 KÖK	25	24	TILL
2 HALL	24.73	24	TILL
3 BADRUM	26.73	24	TILL
4 DATORRUM	24.88	24	FRÅN
5 SOVRUM 1	24.00	24	TILL
6 SOVRUM 2	23.9	24	FRÅN
7 KÄLLARE	24.7	24	TILL
8 VINDEN	38	29	TILL

B = ÅTERSTÄLL LARM
 C = NYA GRÄNSVARDEN

Bild 53. Rubrikerna för mätpunkterna ligger i datasatser och kan därför lätt ändras.

MÄTPUNKT	TEMP	GRÄNS	STATUS
1 KÖK	24.9	24	TILL
2 HALL	24.1	24	TILL
3 BADRUM	26.73	24	TILL
4 DATORRUM	24.73	24	FRÅN
5 SOVRUM 1 > FEL		(KORTSLUTNING)	
6 SOVRUM 2	23.8	24	FRÅN
7 KÄLLARE	24.6	23	TILL
8 VINDEN	38.4	29	TILL

B = ÅTERSTÄLL LARM
 C = NYA GRÄNSVARDEN

Bild 54. Om fel uppstår i mätsystemet ges felmeddelande.

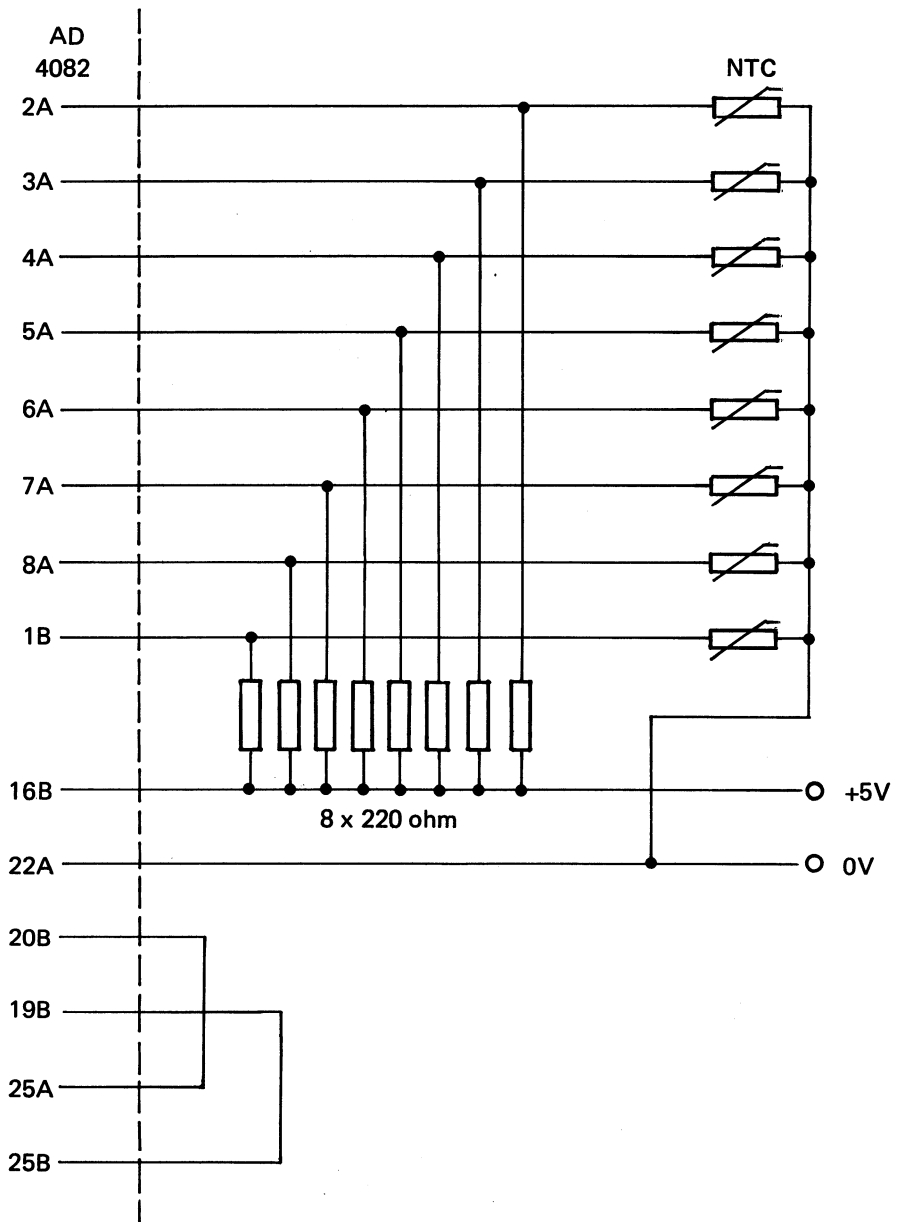


Bild 55. Kopplungsschema.

```

4 REM
10 REM
20 REM *****
30 REM ** PROGRAM FÖR KONTROLL **
40 REM ** AV TEMPERATUR I ATTA **
50 REM ** OLIKA MÄTPUNKTER MED **
60 REM ** AD OMVANDLARE 4082 **
70 REM ** FÖR STYRNING AV **
80 REM ** RELÄKORT 4007 **
90 REM *****
100 REM ** M A D E B Y **
110 REM ** STEFAN ANDERSSON **
120 REM ** JAN SMEDBERG **
130 REM ** VASAGYMNASIET ARBOGA **
140 REM *****
150 F=220 : REM FAST MOTSTAND =220 OHM
160 R1=100 : REM NTC 100 OHM (25 GRADER)
170 K=5000 : REM KONSONANT FÖR NTC
180 A1=8 : REM ANTAL MÄTPUNKTER 1-8
190 A=INP(7) : ; CHR$(12)
200 REM GRADER CELCIUS
210 DEFFNT(T)=K/(K/298-LOG(R1/M1))-273*A
220 ; "ANGE GRÄNSVÄRDET FÖR VARJE MÄTPUNKT"
230 ; "MÄTPUNKT GRÄNSVÄRDE"
240 RESTORE : FOR LX=1 TO A1 : READ M#
250 ONERRORGOTO 250 : ; M#;TAB(18) : INPUT G(LX)
260 NEXT LX : ; CHR$(12) : ONERRORGOTO 0
270 REM MÄT REFERENSSPÄNNING KANAL 31
280 OUT 1,2,2,31,3,0
290 U1=(INP(1)*256+INP(0))*5/4095
300 IF U1<4.75 GOTO 660
310 REM MÄT TEMPERATUR
320 RESTORE : FOR LX=1% TO A1
330 ; CUR(1,0)" MÄTPUNKT TEMP"
340 ; CUR(1,20)" GRÄNS STATUS"
350 ; CUR(13,0)" Å = ÅTERSTÄLL LARM "
360 ; " G = NYA GRÄNSVÄRDEN "
370 FOR I=1 TO 5
380 A=INP(56) : IF A=199% OR A=231% GET A# : GOTO 190
390 IF A=221% OR A=253% : CHR$(12) : OUT 58,0
400 OUT 1,2,2,LX,3,0
410 R=INP(1)*256+INP(0)
420 IF R>3500 OR R<50% 600
430 U=R*5/4095
440 M=U*F/(U1-U)
450 D=D+M
460 NEXT I
470 M1=D/5
480 IF M1<0% 570
490 T(LX)=K/(K/298-LOG(R1/M1))-273
500 T(LX)=INT(T(LX)*10+.5)/10
510 ; CUR(LX+1,13)SPACE$(27)
520 READ M# : ; CUR(LX+1,3)M#
530 ; CUR(LX+1,0)LX;CUR(LX+1,13)T(LX)
540 ; CUR(LX+1,22)G(LX)
550 GOSUB 720
560 D=0
570 NEXT LX
580 GOTO 270
590 REM LARM
600 ; CUR(LX+1,13)SPACE$(27)
610 ; CUR(LX+1,12)"> FEL " ;
620 REM FEL PÅ GIVARE
630 OUT 6,43 : OUT 58,32 : READ M#
640 IF R<50 : " (KORTSLUTNING)" : GOTO 570
650 ; " (AVBROTT)" : GOTO 570
660 REM FEL REFERENSSPÄNNING
670 OUT 6,43 : OUT 58,32 : ; CUR(11,1)
680 ; "FÖR LÅG REFERENSSPÄNNING" : GOTO 320
690 DATA 'K&K','HALL','BADRUM','DATORRUM'

```

```
700 DATA 'SOVRUM 1', 'SOVRUM 2', 'KÄLLARE'  
710 DATA 'VINDEN'  
720 REM RELÄ STYR.  
730 ; CUR(L%+1,32) ; S$(0)="FRAN" ; S$(1)="TILL"  
740 OUT 1,7  
750 IF T(L%)>G(L%) S%=1% ELSE S%=0%  
760 ; S$(S%)  
770 IF S%=1 OUT 0,L%+128 ; RETURN  
780 OUT 0,L% ; RETURN
```

IEC-bussen

Att med en dator kunna samla in mätvärdesinformation samt att få instrument att kommunicera med varandra i ett automatiskt testsystem är i korthet vad man använder IEC-bussen till. 1975 fastställde IEC hur den databuss skulle se ut som möjliggjorde sammankoppling av upp till 15 instrument med en total kabellängd mellan instrumenten på 20 meter. IEC-busskablar, som förbinder datorsystemet med respektive instrument, är försedda med såväl han- som honkontakter som är så monterade att kablarna kan träs på varandra ("piggy back").

Kärt barn har många namn — IEC-bussen är synonym med IEEE 488, HBIB och GPIB. En av orsakerna till standardiseringen av instrumentbussen har varit att man i samma mätuppkoppling skall kunna blanda instrument av olika fabrikat. Instrument som kan vara aktuella i ett automatiskt mätsystem kan vara allt från enkla standard instrument, exempelvis en DMM, till de mest sofistikerade instrument, exempelvis en spektrumanalysator.

En förutsättning för att kunna ansluta ett sådant instrument till instrumentbussen är att instrumentet är IEC- eller IEEE-busskompatibelt, dvs försett med en kontakt som möjliggör anslutning till bussen. Bland instrumenttillverkarna har standarden snabbt blivit accepterad varför det idag finns ett stort utbud av IEC-busskompatibla instrument. Bland de instrumenttillverkare som har hunnit längst märks Hewlett Packard, Rodhe & Schwarz, Fluke och Philips.

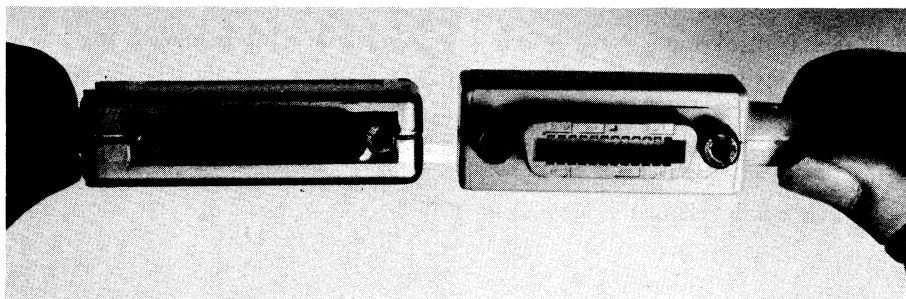


Bild 56. Till vänster den 25-poliga "Cannon D"-kontakten som är IEC-standard. Till höger den 24-poliga "Micro Ribbon"-kontakten som är IEEE 488 standard. Med få undantag använder de flesta instrumenttillverkare den 24-poliga "Micro Ribbon"-kontakten.

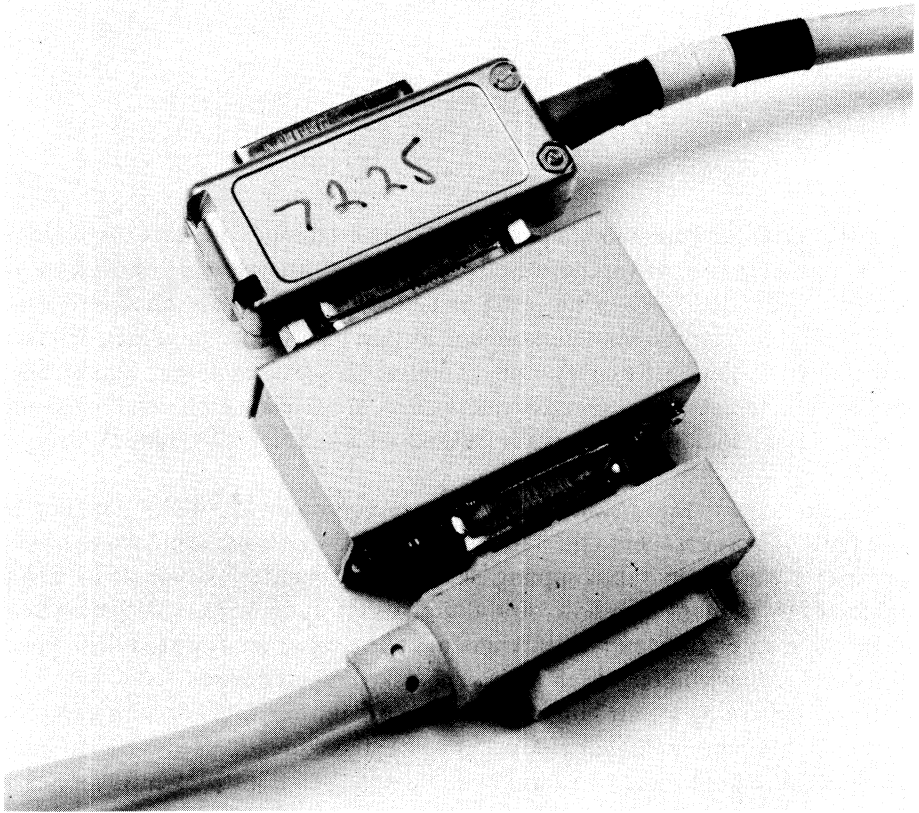


Bild 57. För att åstadkomma anpassning mellan olika kontaktstandard kan en reducering lätt tillverkas.

Hos dessa förekommer två olika kontaktertyper — nämligen IEC och IEEE-standard. Skillnaden ligger i val av kontakt och ledningsplacering. Hewlett Packard, Rodhe & Schwarz samt Fluke använder IEEE-kontakten som är 24-polig medan Philips använder en 25-polig kontakt som är IEC-standard. Sammankoppling av dessa båda system kan dock ske genom att använda en reducering IEC-IEEE.

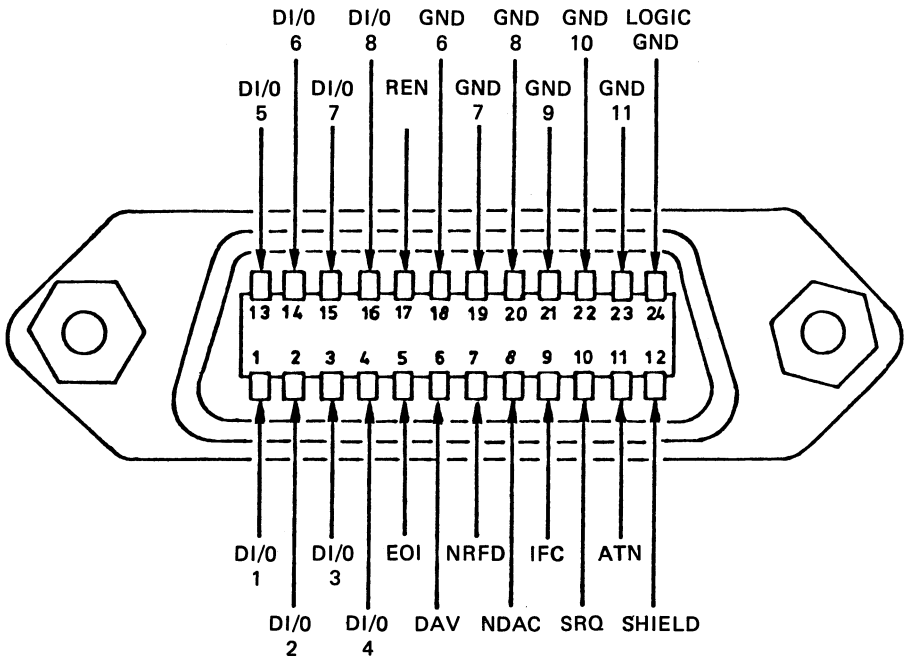


Bild 58. Ledningsplacering enligt IEE-488 standard.

Bussens struktur

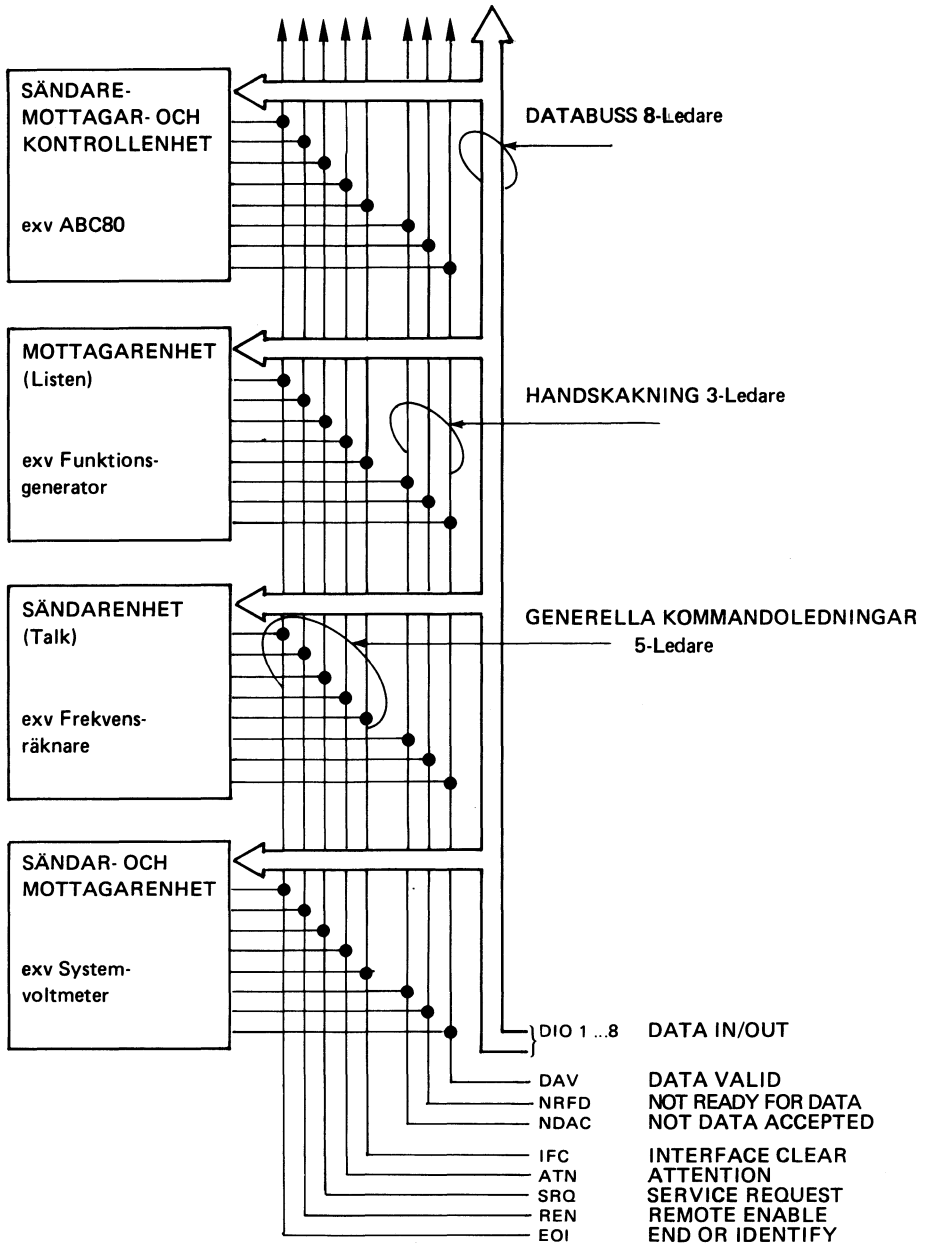


Bild 59. IEC-bussens utformning.

IEC-bussen består av 16 ledningar fördelade på 8 dataledningar för in- och utmatning av data, tre dataöverföringsledningar för handskakning samt fem ledningar för busshantering. De instrument eller enheter som är anslutna till bussen skall innehålla sådana funktioner att de kan utgöra minst en av följande tre funktioner.

Listener dvs fungera som mottagarenhet av data och kommando.

Talker dvs fungera som sändarenhet av data och kommando.

Controller dvs en enhet som kan styra dataflödet på bussen.

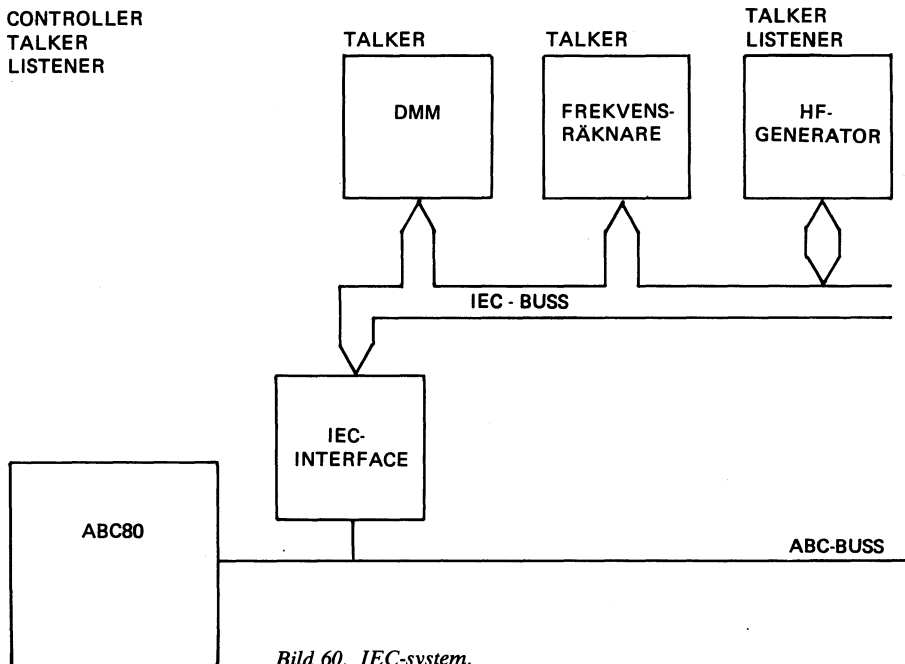


Bild 60. IEC-system.

En DVM är ett exempel på ett instrument som måste ha en talker-funktion medan en systemvoltmeter dessutom måste ha listeners funktion för automatiskt områdesval. En HF-generator är exempel på ett instrument som måste ha listeners-funktion för automatisk ändring av frekvens, modulation och utspänning. En ABC80 tillsammans med ett IEC-interface fungerar som controller men också som talker och listeners.

Högst en talker kan vara verksam åt gången, medan flera listeners samtidigt kan ta emot information. Alla enheter på bussen har en adress som kontrollern skickar ut och därmed klargör för enheterna vilka som skall vara talker resp. listeners. Instrument som kan vara både talker och listners har två adresser för val av funktion.

Adressval

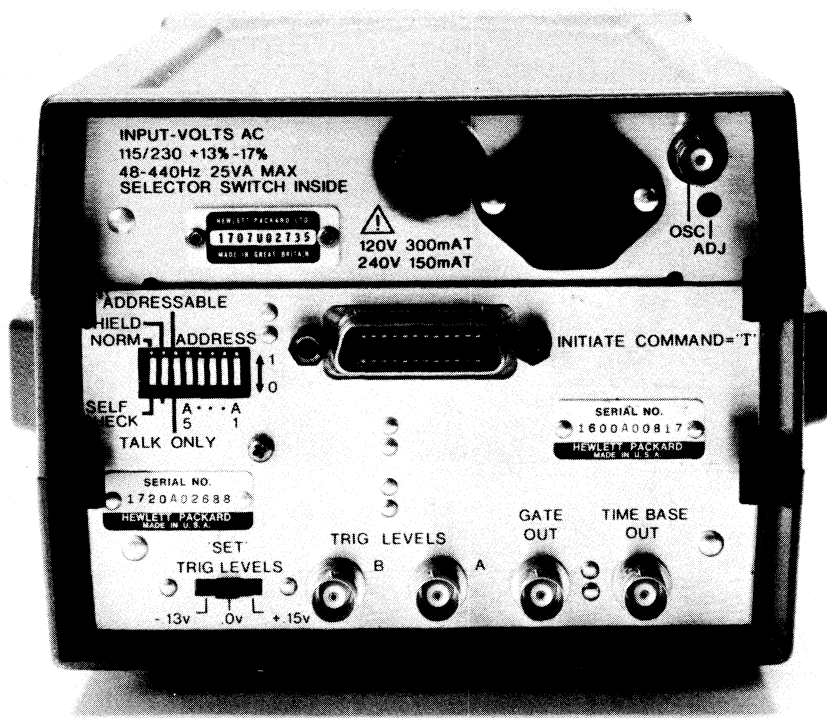


Bild 61. Adressomkopplaren är här placerad tillsammans med IEEE-busskontakten. Adressomkopplarens placering gör att dess inställning skyddas av bussledningen.

Alla till bussen inkopplade enheter tilldelas en egen adress. Detta sker med en omkopplare, oftast placerad tillsammans med busskontakten på instrumentetens baksida. Med adressomkopplaren ställs de fem lägsta bitarna i den sju bitar långa adressen in. De två högsta bitarna bestämmer om enheten skall ha en sändande eller mottagande funktion.

ADDRESS SWITCH					Talk Address	Listen Address
A ₅	A ₄	A ₃	A ₂	A ₁		
0	0	0	0	0		SP
0	0	0	0	1	A	!
0	0	0	1	0	B	"
0	0	0	1	1	C	#
0	0	1	0	0	D	\$
0	0	1	0	1	E	%
0	0	1	1	0	F	&
0	0	1	1	1	G	'
0	1	0	0	0	H	(
0	1	0	0	1	I)
0	1	0	1	0	J	*
0	1	0	1	1	K	+
0	1	1	0	0	L	,
0	1	1	0	1	M	-
0	1	1	1	0	N	.
0	1	1	1	1	O	/
1	0	0	0	0	P	0
1	0	0	0	1	Q	1
1	0	0	1	0	R	2
1	0	0	1	1	S	3
1	0	1	0	0	T	4
1	0	1	0	1	U	5
1	0	1	1	0	V	6
1	0	1	1	1	W	7
1	1	0	0	0	X	8
1	1	0	0	1	Y	9
1	1	0	1	0	Z	:
1	1	0	1	1	Å	;
1	1	1	0	0	ä	<
1	1	1	0	1	å	=
1	1	1	1	0	(>
1	1	1	1	1)	?

BIT b7 b6 b5 b4 b3 b2 b1
 TALK ADDRESS 1 0 A⁵ A⁴ A³ A² A¹
 LISTEN ADDRESS 0 1 A⁵ A⁴ A³ A² A¹

Av tabellen för adresskoder framgår, att om en enhet ges adressen R (i ASCII-form) så får instrumentenheten en sändande funktion, medan samma

enhet får en mottagande funktion då den tilldelas adressen 2. ABC80 har via IEC-interfacet 4025 fasta adresser, nämligen

TALKER adress U
LISTENER adress 5

När ett ?(UNL=unlistener) läggs ut, nollställs alla tidigare listeneradresser. En talkadress nollställs då en ny talkadress läggs ut. För att i program kunna bestämma in- repektive urkoppling av instrumentenheter skulle en programsekvens kunna vara följande

30 CMD "?U7"+CHR 8 (8)
40 CMD "?W5"

På rad 30 nollställs alla listeneradresser av ?. ABC80 beordras som talker och enheten med listeneradressen 7 beordras som mottagare. Genom instruktionen CHR 8 (8) ges order om start av mätning. På rad 40 nollställs på nytt alla listeneradresser och den enhet, som på föregående rad fick order att starta mätning, övergår nu i talkfunktion medan ABC80 blir mottagare genom listeneradressen 5.

Programmering av IEC-bussen

Programmeringen av IEC-bussen är till viss del beroende av de instrument, som kopplas till bussen. Studera därför manualen för respektive instrument innan programmeringen påbörjas. För hanteringen av IEC-bussen fordras att IEC-interfacet kompletteras med ett IEC-drivprogram, som för IEC-interface 4025 placeras på PROM-kortet i flexskiveminnet expansionsutrymme. För IEC-interface ABC IEC sitter det aktuella drivprogrammet på interfacekortet.

Börja programmeringen med att reservera en buffert i ABC80 för IEC genom att skriva

```
10 OPEN "IEC:" ASFILE X
```

där X är ett filnummer som kan variera mellan 1 och 255. Med denna sats initieras IEC-interfacet. Vidare sätts ABC80:s talkadress till U och listeneradress till 5. Samma filnummer används då man vill avsluta kommunikationen med bussen. Stängning av filen sker med

```
1000 CLOSE X
```

Kommando och data överförs på IEC-bussen genom att skriva t ex

```
60 CMD A $\square$  , B $\square$ 
```

Innehållet i strängarna sänds ut i ASCII-form. A \square är en textsträng som definierar talker och listener. B \square är en kommandotextsträng. A \square :en kan utgöras av t ex "?U6" där

? =nollställning av alla linsteners (UNL)

U=ABC80 beordras som talker

6 =instrument med adress 6 beordras som listener

B \square kan bestå av ett uttryck som t ex "BCD1F2A". Uttrycket är speciellt för ett visst instrument och innehåller kommandon till instrumentet. Dessa kommandon kan bestå av uppgifter om områdesval, funktionsslag, antal mätningar per tidsenhet, start av mätning osv.

Utskrift av mottagna data från IEC-systemet sker med

```
PRINT IEC $\square$  (13)
```

Satsen skriver i detta fallet ut 13 tecken. Utskriftens karaktärsantal kan variera

från instrument till instrument. För Hewlett Packards digitala multimeter 3438A är utskriftsformatet 13 karaktärer långt och uppdelat enligt

+D.DDD	E±D	,	F	CR	LF
Display	Exponent	Delimiter	Function Code	Carriage Return	Line Feed
Input Value			Function	Code	
			DCV	1	
			ACV	2	
			DCI	3	
			ACI	4	
			OHMS	5	

Om den digitala multimetern visar ett värde av t ex +17,90 volt blir en typisk utskrift +1,790 E+1,1. Den sista siffran, som i detta fallet är en etta, talar om att värdet avser en likspänningsmätning (DCV). Om mätvärdet, utöver utskrift, skall användas, för exempelvis beräkningar måste värdet överföras och lagras i en variabel. Detta kan ske med

$$V_n = IEC_n \quad (13)$$

Inkoppling av IEC-utrustningen

ABC80 kan med stor fördel användas som mät dator (IEC-bussdator) under förutsättning att ABC80 kompletteras med ett IEC-buss-interface. Det finns på marknaden två IEC-buss-interface för ABC80, nämligen

Sattcos IEC-buss-system som består av:

IEC-buss-interface 4025

IEC-drivprogram i PROM 9702

IEC-busskabel 7225

Scandia Metrics

ABC-IEC-interface

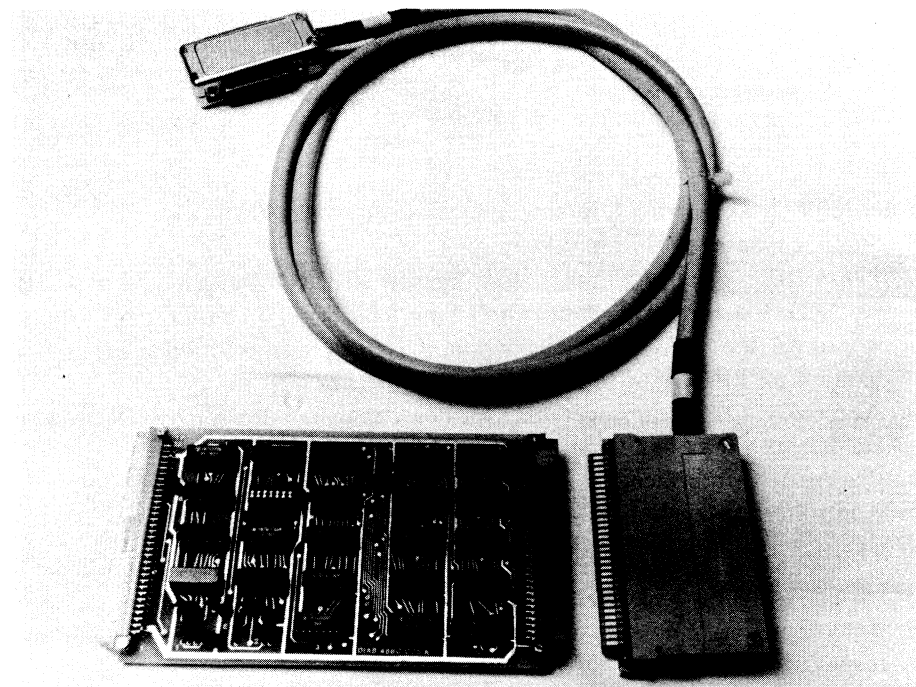


Bild 62. IEC-buss interface och busskabel.

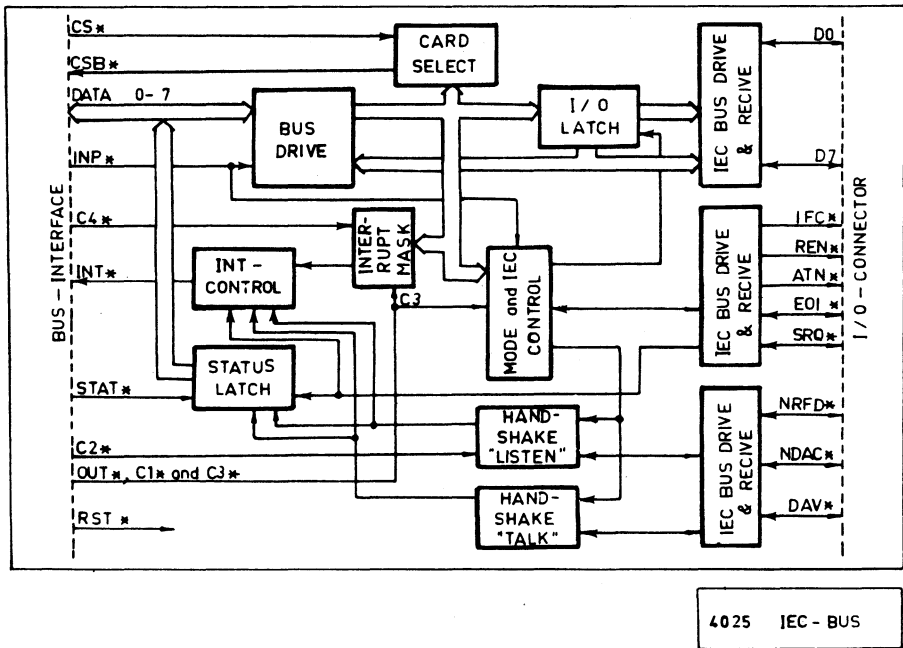


Bild 63. Blockschema IEC-interface 4025.

I det följande beskrivs inkopplingen av IEC-interface 4025.

1. Slå av spänningen till flexskiveminnet.
2. Placera IEC-drivprogrammet, som ligger i en PROM-kapsel, på plats 1B på PROM-kortet 3032. Detta sitter längst till höger i DataDiscens expansionsutrymme. Används flexskivminne ABC placeras drivprogrammet i håller 4 på PROM-kortet.
3. Adressbygla IEC-interfacet 4025 för den decimala adressen 49. Detta sker genom att klippa byglingarna 1, 5 och 6.
4. Placera IEC-interfacet på valfri plats i expansionsutrymmets I/O-del.
5. Kontrollera inkopplingen av IEC-interfacet genom att skriva

OUT 1,49

varvid lysdioden på kortets framkant skall tändas.

6. Tilldela mätsystemets instrument talk- och listeneradresser med hjälp av respektive instruments adressomkopplare.
7. Sammankoppla instrumenten med IEC-interfacet. Detta kan ske i kedje- eller stjärnform. Övergångar mellan IEC-kontakt bestyckade instrument och sådana som har IEEE-kontakter ordnas med reducering.

Mätning med IEC-buss

Som tidigare nämnts finns det på marknaden ganska många IEC-busskompatibla instrument. De flesta ligger i en prisklass, som gör att de troligen inte kommer att vara så vanligt förekommande på t ex skolor som man skulle önska. Men undantag finns, bl a Hewlett Packards DMM 3438A samt HP:s mätsystem 5300.

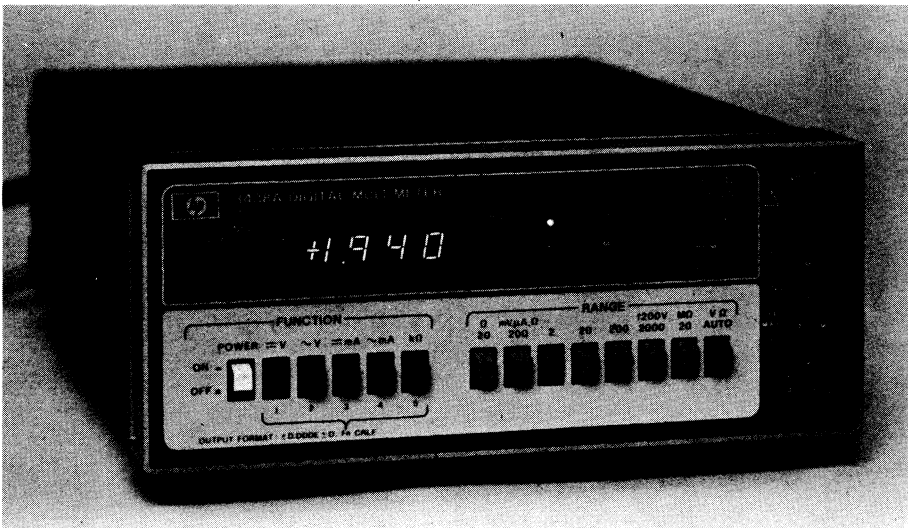


Bild 64. Hewlett Packards digitala multimeter 3438A som är ett instrumentbusskompatibelt instrument.

DMM 3438A är en traditionell digital multimeter som är anslutningsbar till ett IEC-buss-system. Begränsningen ligger i att mätområdesskiftning och mätfunktionsskiftning måste göras manuellt. Detta beror på att instrumentet endast har en talk-funktion (talk only mode). Vill man med datorn styra även mätområde och mätfunktion måste man välja en systemvoltmeter. En sådan medger också mångdubbelt fler mätningar per tidsenhet.

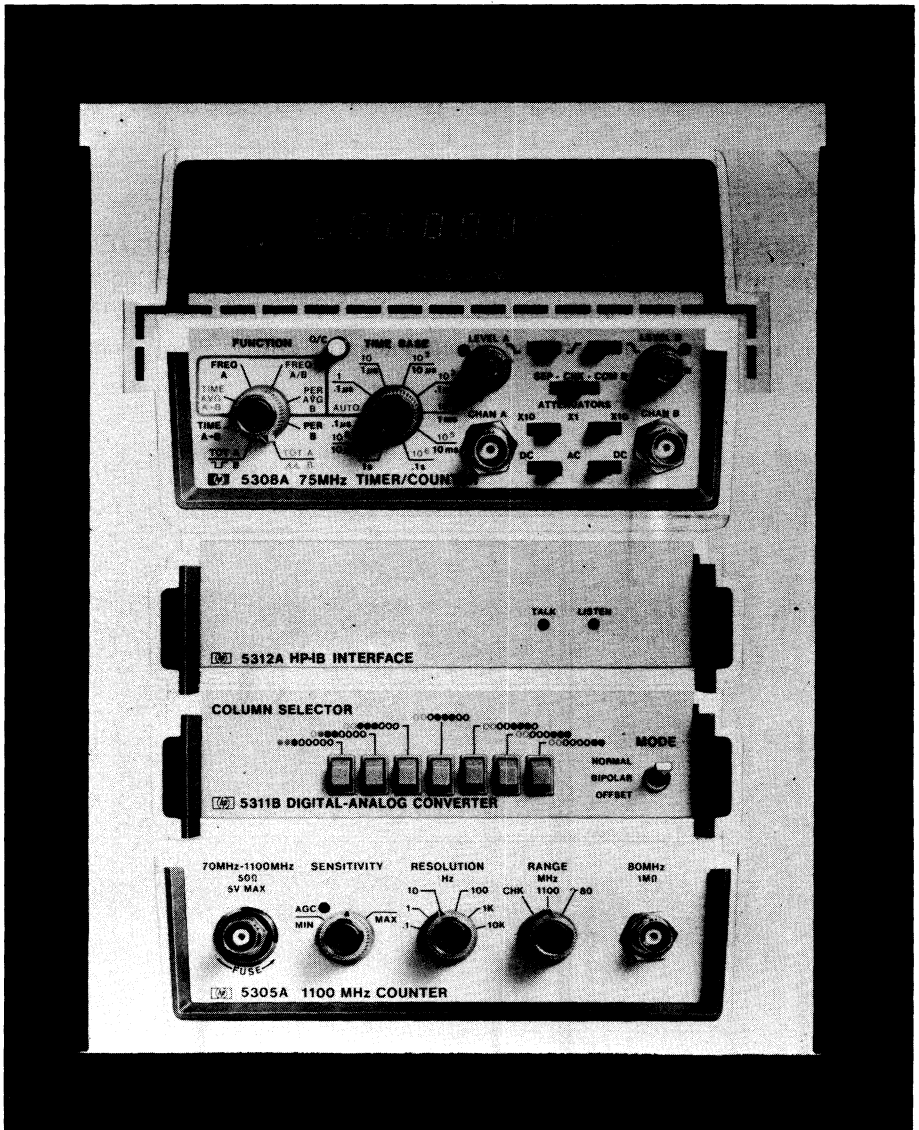


Bild 65. Enheten i mitten är ett HP-IB-interface vilket gör instrumentsystemet anslutningsbart till en IEC-buss (Hewlett Packard).

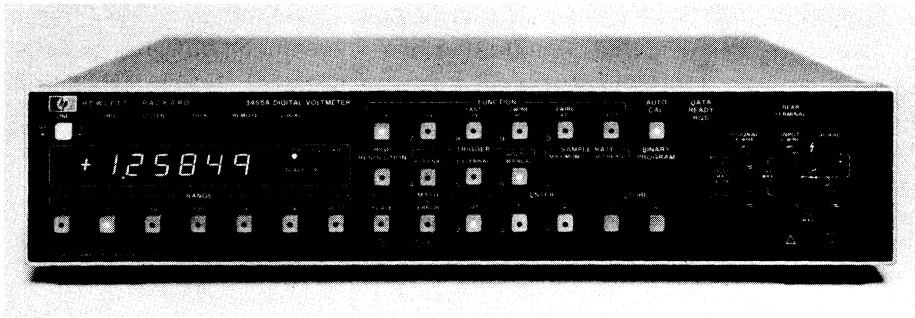


Bild 66. Systemvoltmeter 3455A Hewlett Packard.

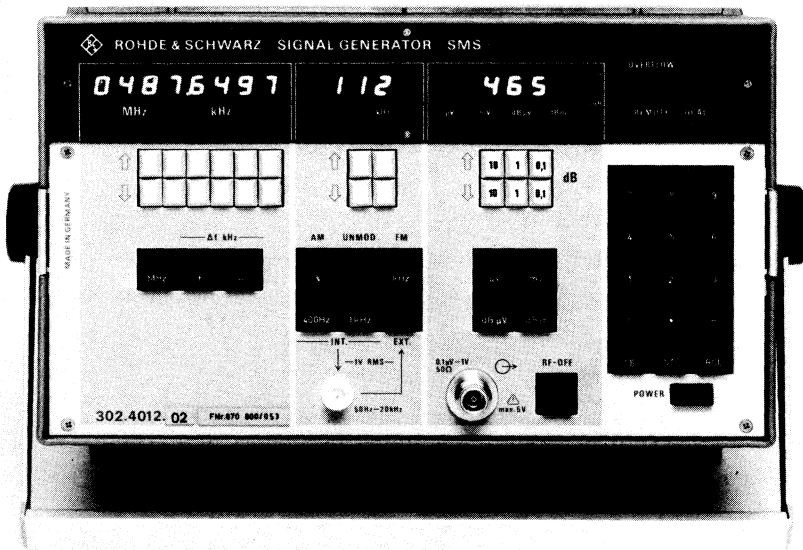


Bild 67. HF-generator SMS är en IEC-kompatibel AM/FM-generator för frekvensområdet 0,4–1040 MHz. (Rodhe & Schwarz)

Under locket på DMM 3438A sitter adressvalsomkopplaren. Tildela instrumentet en talkadress t ex W och skriv därefter följande program.

```
10 OPEN "IEC:" ASFILE 1
20 CMD "?W5"
30 V$=IEC$(11)
40 CMD "? "
50 ; CUR(12,12)V$ " : GOTO 20
```

På rad 10 initieras IEC-buss-utrustningen bl a genom att kortvalsadress 49 läggs ut av drivprogrammet. Rad 20 nollställer med ? eventuella tidigare listneradresser, varefter DMM beordras till talker med adress W och ABC80 till listner med adress 5. Informationen från DMM tilldelas på rad 30 en variabel V□, som skrivs ut på bildskärmen (rad 50), varefter ett nytt värde hämtas in. Ändra programmet enligt följande och iakttag ändringen av utskriften.

```

10 OPEN "IEC:" ASFILE 1
20 CMD "2W5"
30 V$=IEC$(11)
40 CMD "?"
50 ONERRORGOTO 20 : V=VAL(MID$(V$,2,8))
60 ; CUR(12,12)V" " : GOTO 20

```

Den väsentliga skillnaden ligger på rad 50 där uttrycket

$$V = \text{VAL}(\text{MID}(V\Box, 2, 8))$$

omvandlar innehållet i V□ till ett numeriskt värde. Av V□-ns ursprungliga 11 karaktärer sker nu utskrift från och med den andra positionen med åtta karaktärer. Den sista siffran är en funktionskod, som anger funktionsväljarens läge.

Funktion	Code
DCV	1
ACV	2
DCI	3
ACI	4
OHMS	5

Genom att komplettera programmet enligt följande kan direkt utskrift ske av aktuellt mätfunktionsområde. När mätningarna är avslutade stängs IEC-filen genom att skriva

CLOSE 1

som kommando.

```

10 REM RELÄ4007
20 REM STYRNING AV 8 RELÄ
30 A=INP(7)
40 OUT 1,7
50 A$='12345678'
60 B$='!'"£$%&/'<'
70 T$=CHR$(INP(56))
80 S=INSTR(1,A$,T$)-1
90 V=INSTR(1,B$,T$)-1
100 IF S>-1 THEN OUT 0,S+128
110 IF V>-1 THEN OUT 0,V
120 IF T$='A' THEN OUT 4,0
130 GOTO 70
140 END

```

IEC-bussmätning av frekvensgång

Av bild 68 framgår schemat för ett IEC-buss-system som möjliggör halvautomatisk mätning av frekvensgången hos en LF-förstärkare.

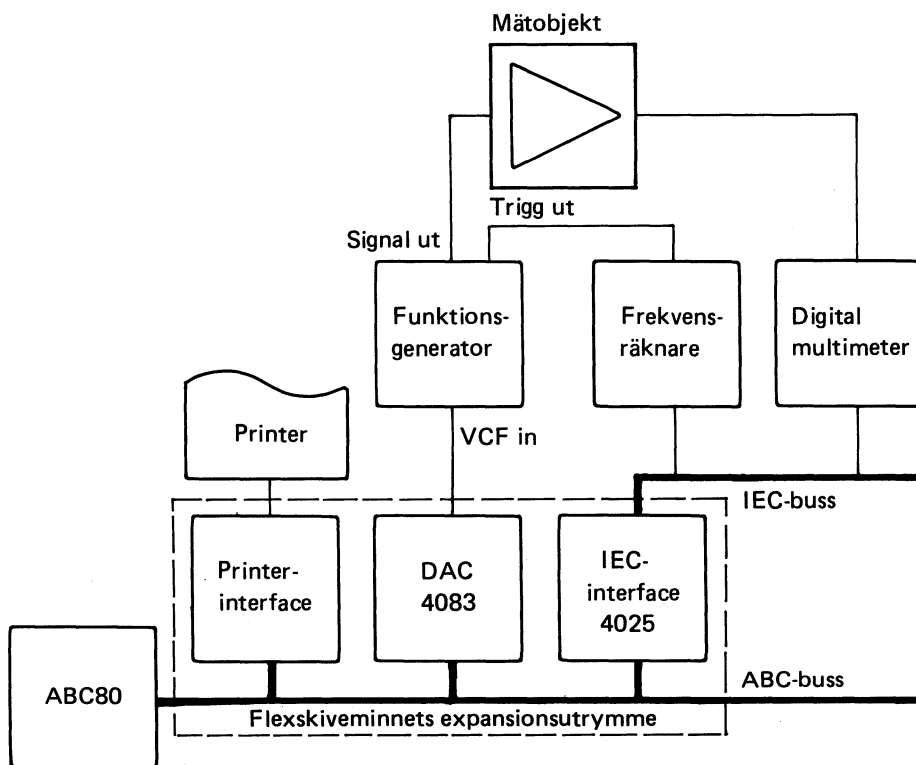


Bild 68. IEC-uppkoppling för mätning av frekvensgången hos en förstärkare.

Till ABC-bussen är anslutet en DAC och ett IEC-interface. DAC-n styr en funktionsgenerator via dess VCF-ingång. Signalutgången på funktionsgeneratoren är ansluten till LF-förstärkarens signalingång. Till IEC-bussen är en DMM och en frekvensräknare anslutna. För att förse frekvensräknaren med en tillfredställande amplitud ansluts dess ingång till funktionsgenerators trigg-



Bild 69a. ABC80 som IEC-dator. Till IEC-bussen är de båda övre instrumenten inkopplade. Funktionsgeneratoren längst ner i mitten styrs från en DAC. Mätning görs av frekvensgången på receiveern till höger bilden.

utgång. I kopplingen är DAC-n dels byglad för 10 volts referensspänning på kanal 1, dels byglad för kortvalsadressen 3. Programmet som framgår av sid 110–112 är till stor del självinstruerande. Instruktionerna ges fortlöpande under programexekveringen på bildskärmen med PRINT-satser. Dessa är kompletterade i programlistan med REM-satser.

På raderna 1430—1550 finns en subrutin för reglering av DAC-ns utspänning. En spänningsändring på DAC-ns utgång medför en motsvarande frekvensändring hos funktionsgenerators utspänning. Detta under förutsättning att generators frekvensområdesväljare står i rätt läge. DAC-n kommer att öka sin utspänning i steg om 0,5 volt. Efter varje sådan stegökning av DAC-ns utspänning kopplas frekvensräknaren in och läser funktionsgenerators frekvens. Om det inlästa värdet på frekvensen F_1 har nått en storlek, som överstiger det värde som mätfrekvensen F skall ha, kommer DAC-ns utspänning att minska. Samtidigt som nu spänningen går i negativ riktning ändras också steglängden till 0,05 volt. Insvängningsförloppet har därmed gått in i en finregleringsfas. När $F_1 < F$ ändras på nytt steglängden till 0,005 volt samtidigt som DAC-ns utspänning på nytt ökar. Detta insvängningsförlopp pågår tills $F_1 = F$. Eftersom inläsning av frekvensen sker efter varje spänningsändring hos DAC-n kommer inställningstiden att bli ganska lång (ca 10 s). Stor noggrannhet erhålls i gengäld.

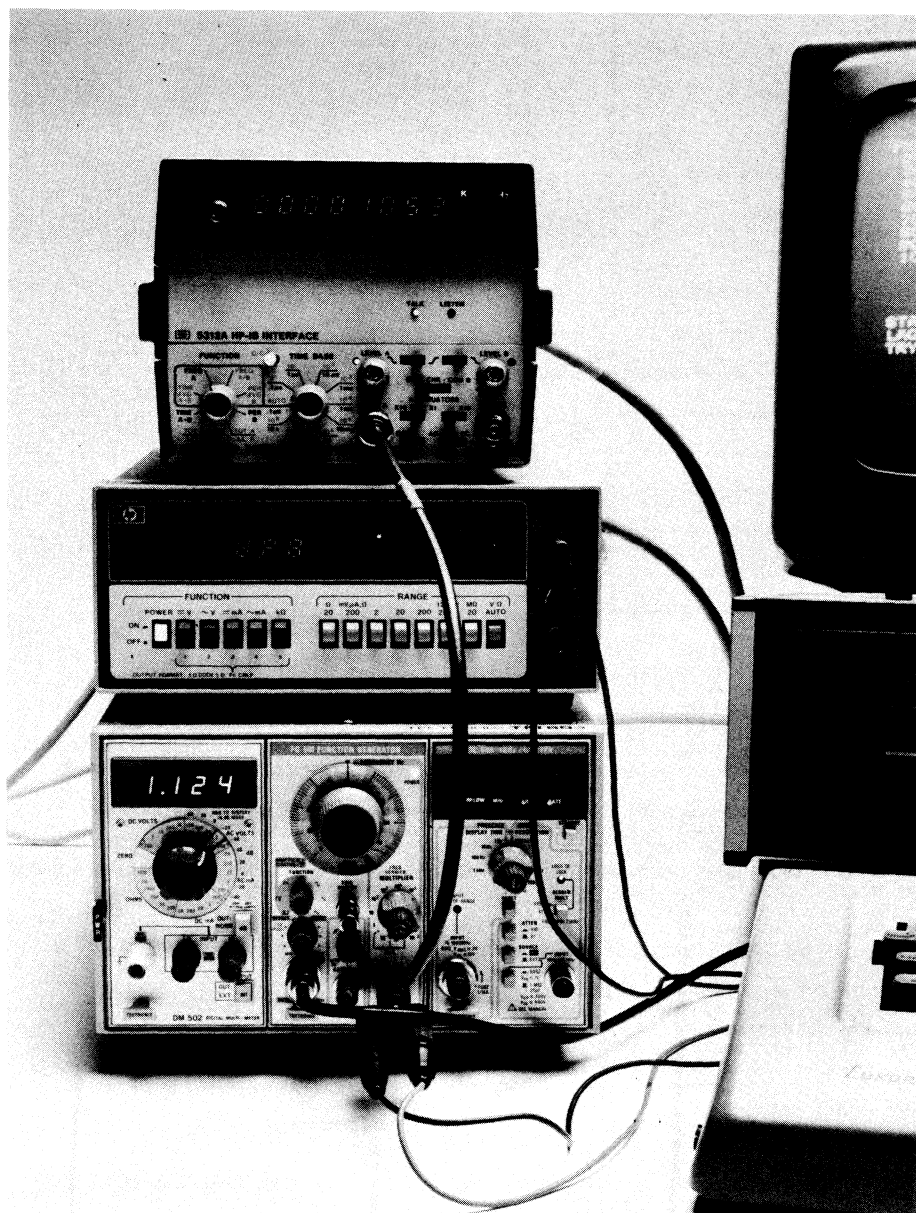


Bild 69b. Många bedömare är överens om att 80-talets mätteknik är instrument-busstillämpningar. Antalet IEC-busskompatibla instrument ökar snabbt. Behovet av en prisvärd IEC-busstillämpning har styrt valet av instrument som här omfattar en Frekvens-räknare med HP-IB interface i Hewlett-Packards 5300-serie, en DMM 3438A Hewlett-Packard samt en Funktionsgenerator Tektronix FG501.

```

100 REM *****
110 REM * *
120 REM * MÄTNING AV EN FÖRSTÄRKARES *
130 REM * *
140 REM * FREKVENSGANG MED IEC BUSS *
150 REM * *
160 REM * OCH DAC *
170 REM * *
180 REM * MED DATABOARD KORTEN *
190 REM * *
200 REM * 4025 & 4083 *
210 REM * *
220 REM * INSTRUMENT *
230 REM * *
240 REM * HP'S FREKVENSRÄKNARE 5308A *
250 REM * *
260 REM * HP'S MULTIMETER 3438A *
270 REM * *
280 REM * TEKTRONIX FUNK-GEN FG 501 *
290 REM * *
300 REM *****
310 REM
320 REM *****
330 REM * *
340 REM * STEFAN ANDERSSON *
350 REM * *
360 REM * BJÖRN GRANBERG *
370 REM * *
380 REM * ARBOGA 1980-05-06 *
390 REM * *
400 REM *****
410 REM _
420 REM START
430 DIM V(50),D(50) : OPEN "IEC:" ASFILE 1 : OPEN "PR:" ASFILE 2 : J2=0% : I2=0% : J=1 : CHR$(12)
440 : "MÄTNINGSÄMPLING MED IEC-BUS" : : STRING$(25,45) :
450 : "FÖLJANDE INSTRUMENT BEHÖVS" : :
460 : "HP:5 DIGITAL MULTIMETER 3438A" : : "HP:5 FREKVENSRÄKNARE 5308A"
470 : "TEKTRONIX FUNKTIONSGENERATOR FG 501" : :
480 : CUR(10,0)"FÖLJANDE DATABOARD KORT BEHÖVS" : : : "D/A OMVÄNDLARE 4083" : : "IEC-BUS ANPASSNING 4025"
490 GET G$ : : CHR$(12)
500 : "INKOPPLING AV INSTRUMENTEN OCH" : : "DATABOARD KORTEN" : : : "BYGLA DAC KORTET FÖR 10 VOLT UT"
510 : "PÅ KANAL 1 " : : "ANSLUT DAC KORTETS KANAL 1 TILL" : : "FUNKTIONSGENERATORNS VCF INGANG" : :
520 : "ANSLUT FUNKTIONSGENERATORNS TRIGG-" : : "UTGANG TILL KANAL A PÅ FREKVENNS-" : : "RÄKNAREN" : :
530 : "ANSLUT FUNKTIONSGENERATORNS UTGANG" : : "TILL FÖRSTÄRKARENS INGANG" : : : "TRYCK PÅ RETURN NÄR DU ÄR KLAR" : :
540 GET G$ : IF G$=CHR$(13) 540 ELSE : CHR$(12)
550 REM FREKVENSRÄKNARENS TALADRESS A
560 REM MULTIMETERNS TALADRESS W
570 C=3 : REM DAC KORTETS CARDSELECT 3
580 REM _
590 REM INFORMATION OCH 1000 HZ 0 DB MÄTNING
600 : "FÖRSTA MÄTNINGEN SOM SKA GÖRAS ÄR" : : "FÖRSTÄRKNINGEN VID 1000 HZ" : : : :
610 : "STÄLL FREKVENSRÄKNAREN I LÄGE FREQ A OCH 'TIME BASE' I LÄGE AUTO" : :
620 : "STÄLL FUNKTIONSGENERATORNS FREKVENSSKALA I LÄGE 1 OCH OMRÅDESKOPPLAREN I LÄGE FREKVENNS * 100" : :
630 : "STÄLL MULTIMETERNS FUNKTIONSKOPPLARE I LÄGE AC-VOLT AUTO" : :
640 : "KOPPLA MULTIMETERN TILL FÖRSTÄRKARENS INGANG." : : : "TRYCK PÅ RETURN NÄR DU ÄR KLAR" : : : GET G$
650 : CUR(6,0)SPACE$(50)
660 GOSUB 1570 : GOSUB 1620 : REM NOLLSTÄLLNING AV DAC OCH INSTRUMENT
670 X=100 : F=1000 : GOSUB 1430 : GOSUB 1660 : R1=V : REM LÄS INSPÄNNINGEN TILL R1
680 : CHR$(12,10,10,10)"INSPÄNNINGEN ÄR "R1" VOLT "
690 : : "KOPPLA NU MULTIMETERN TILL FÖRSTÄRKARENSUTGANG" : : : "TRYCK PÅ RETURN NÄR DU ÄR KLAR" : : : GET G$
700 GOSUB 1660 : R2=V : REM LÄS UTSPÄNNINGEN TILL R2
710 R=INT((20+LOG10(R2/R1))*100+ 5)/100 : REM R= FÖRSTÄRKNINGEN I DB VID 1000 HZ AVRUNDAT TILL TVÅ DECIMALER
720 : CHR$(12)"INSPÄNNING "R1" VOLT " : : "UTSPÄNNING "R2" VOLT " : : "FÖRSTÄRKNING "R" DB " : : "FREKVENNS 1000 HZ"
730 : CUR(8,0)
740 : "DESSA VÄRDEN KOMMER ATT ANVÄNDAS" : : "SOM REFERENS"
750 : "DÄRFÖR FÅR INTE GENERATORNS ELLER" : : "FÖRSTÄRKARENS UTSPÄNNINGAR ÄNDRAS" : : "UNDER MÄTNING"
760 : CUR(20,5)"TRYCK PÅ RETUN NÄR DU LÄST KLART"
770 GET G$ : : CHR$(12)CUR(6,0)
780 REM _
790 REM MÄTNINGSLOOP 10 HZ - 100 KHZ

```

```

800 ; "MÄTOMRÅDET ÄR 10KHZ - 100KHZ" ; ; "OM FREKVENSEN ÄR HÖGRE ÄN 1KHZ OCH" ; ; "UTSPÄNNINGEN MINDRE ÄN -40 DB"
810 ; "AVSLUTAS MÄTNINGEN AUTOMATISKT"
820 V=0 : GOSUB 1570 : X=10 : IZ=0% : FOR I=1 TO 4
830 OUT 6,0 : OUT 6,135
840 ; CUR(15,0)"STÄLL GENERATORNS OMRÅDESMOKKOPPLARE I LÄGE FREKVENSN *X* ; ; "TRYCK SEDAN PÅ RETURN" ;
850 GET G$ : IF G$=CHR$(13) 830
860 ; CHR$(12) ; ; "FREKV HZ","UTSP VOLT","DB"
870 GOSUB 1620 : REM LASER FREKVENSRÄKNAREN EFTER ÄNDRING
880 FOR J=1 TO 10 : IF X<10 AND J=1 J=2
890 F=J*X : GOSUB 1430 : GOSUB 1660
900 V(IZ)=V : D(IZ)=INT(((20*LOG10(V/R1)-R)*100+.5)/100) : REM DB VÄRDET FÖR VARJE MÄTNING AVRUNDAT TILL TVÅ DECIMALER
910 IF D(IZ)<-40% AND F>1000% 960
920 ; F,V(IZ),D(IZ)
930 IZ=IZ+1% : NEXT J : X=F : NEXT I
940 REM _
950 REM SAMMANSTÄLLNING AV MÄTVÄRDEN
960 ; CHR$(12)"***** SAMMANSTÄLLNING AV MÄTVÄRDEN *****" ; ;
970 ; ; "1 LISTA PÅ SKÄRMEN" ; ; "2 LISTA PÅ PRINTERN" ; ; "3 GRAFIK" ; ; "4 PLOTTNING PÅ PRINTER"
980 ; "MÄT NY MÄTNING"
990 ; "SLUT PROGRAMAVSLUTNING" ; ; ; "VILKEN RUTIN" ; ; INPUT G$ : ONERRORGOTO 960
1000 IF INSTR(1,G$,"MAT")>0 OR INSTR(1,G$,"MAT")>0% 430
1010 IF INSTR(1,G$,"SLUT")>0 OR INSTR(1,G$,"SLUT")>0% END
1020 IF VAL(G$)>4% OR VAL(G$)<1% 960 ELSE ON VAL(G$) GOTO 1050,1060,1170,1250
1030 REM _
1040 REM LISTA
1050 FZ=0% : GOTO 1070
1060 FZ=2% : REM PRINTERFILLEN!
1070 ; CHR$(12) ; ; FZ;"FREKV HZ","UTSP VOLT","DB"
1080 JZ=0 : X=10 : FOR I=1 TO 4
1090 FOR J=1 TO 10 : IF X<10 AND J=1 J=2
1100 F=J*X : FZF,V(JZ),D(JZ) : IF PEEK(65011)>20% GET G$
1110 JZ=JZ+1% : IF JZ=1% 1130
1120 NEXT J : X=F : NEXT I
1130 ; ; "SLUT PÅ LISTAN" ; ; "TRYCK PÅ RETURN NÄR DU LÄST KLART" ;
1140 GET G$ : IF G$=CHR$(13) 960 ELSE 1140
1150 REM _
1160 REM GRAFISK
1170 ; CHR$(12)CUR(0,6)"DB" : FOR I=1 TO 21 : ; CUR(I,4):(11-I)*3;TAB(7)CHR$(151)"N" : NEXT I
1180 FOR I=19 TO 57 STEP 9 : SETDOT 65,I : NEXT I : ; CUR(21,30)CHR$(134)"HZ"
1190 ; CUR(22,10)"10 100 100 1K 10K 100K"
1200 FOR I=0 TO IZ-1
1210 IF INT(D(I)+.5)<-30 OR INT(D(I)+.5)>30% 1230
1220 SETDOT 34-D(I),I+19
1230 NEXT I
1240 GET G$ : GOTO 960
1250 REM _
1260 REM PLOTTNING PRINTER
1270 ; F2TAB(10)"-4 3 3 3 3 3 2 2 2 2 1 1 1 1 1 1 -DB < > +DB 1 1 1 1 1 2 2 2 2 2 3 3 3 3 4"
1280 ; F2TAB(10)"-0 8 6 4 2 0 8 6 4 2 0 8 6 4 2 0 8 6 4 2 0 2 4 6 8 0 2 4 6 8 0 2 4 6 8 0 2 4 6 8 0"
1290 ; F2"FREKVENSN "-"STRING$(81,43)
1300 JZ=0% : X=10 : FOR I=1 TO 4
1310 FOR J=1 TO 10 : IF X<10 AND J=1 J=2
1320 F=J*X : ; F2F;TAB(10)"*";
1330 IF INT(D(JZ)+.5)>0 : F2TAB(51)"*";
1340 ; F2TAB(51+INT(D(JZ)+.5))"*";
1350 IF INT(D(JZ)+.5)<0 : F2TAB(51)"*+" ELSE ; F2
1360 ; F2TAB(10)"*";TAB(51)"*+"
1370 JZ=JZ+1% : IF JZ=1% 1390
1380 NEXT J : X=F : NEXT I
1390 FOR J=1 TO 20 : ; F2 : NEXT J : GOTO 950
1400 REM
1410 REM ____ SUBROUTINER ____
1420 REM
1430 REM INSTALLNING AV FREKVENSEN F
1440 U=J-1 : W=10 : S=-.5 : GOSUB 1490 : IF F1*1000=F RETURN
1450 U=V : W=0 : S=-.05 : GOSUB 1490 : IF F1*1000=F RETURN
1460 U=V : W=10 : S=-.005 : GOSUB 1490 : IF F1*1000=F RETURN
1470 U=V : W=0 : S=-.0005 : GOSUB 1490
1480 RETURN
1490 FOR V=0 TO W STEP S : GOSUB 1570 : GOSUB 1620 : IF SGN(S)<0% 1510

```

```

1500 IF F1>=F/1000 RETURN ELSE 1520
1510 IF F1<=F/1000 RETURN
1520 NEXT V
1530 ; CHR$(12); "KAN EJ STALLA IN GENERATORN" ; ; "PÅ" F "HZ"
1540 ; "KONTROLLERA ATT OMRÅDESKOPPLAREN" ; ; "STÅR I LÅGE FREKVENNS *X"
1550 ; CUR(20,5) "TRYCK PÅ RETURN" : GET G$ : ; CHR$(12) : GOTO 1430
1560 REM _
1570 REM DAC INSTALLNING TILL V VOLT
1580 V%=V*4095/10
1590 OUT 1,C,0,V%/2,SWAP%(V%) : REM UT KANAL 1
1600 RETURN
1610 REM _
1620 REM LÄSFREKVENNS TILL F1
1630 CMD "?AS" : F#=IEC$(12) : CMD "?"
1640 F1=VAL(RIGHT$(F#,3)) : RETURN
1650 REM _
1660 REM LÄS SP TILL V
1670 CMD "?MS"
1680 V#=IEC$(11) : IF RIGHT$(V#,11)<>"2" ; CHR$(12) "STALL MULTIMETERN I LÅGE AC VOLT" : GOTO 1670
1690 CMD "?"
1700 ONERRORGOTO 1670 : V=VAL(MID$(V#,2,8)) : RETURN

```

Om programraderna 1440—1470 skrivs om till följande lydelse kommer mättiden avsevärt att förkortas.

```

1440 U=J-1 : W=10 : S=.5 : GOSUB 1490 : IF F1*1000<F*1.1 RETURN
1450 U=V : W=0 : S=-.05 : GOSUB 1490 : IF F1*1000>F*.9 RETURN
1460 U=V : W=10 : S=.005 : GOSUB 1490 : IF F1*1000<F*1.1 RETURN
1470 U=V : W=0 : S=-.0005 : GOSUB 1490

```

I programmets början skall en referensmätning göras för att erhålla 0dB vid 1000 Hz. Vid denna mätning är det därför nödvändigt att flytta DMM:s anslutning till förstärkaringsången. Efter registrering av insignalens amplitud vid 1000 Hz beräknas förstärkningen enligt

$$20 * \text{LOG}_{10}(\text{UTSP} / \text{INSP})$$

Värdet lagras i variabeln R och används som 0dB-referens. Detta manuella förfarande kan undvikas om systemet kompletteras med ytterligare en digital multimeter. Efter att ha kopplat DMM tillbaka till förstärkarutgången startar den egentliga mätningen. I en loop på raderna 820—930 läser datorn förstärkarens utspänning vid nio mätfrekvenser/dekad från 10 Hz till 100 kHz. Vid varje mätfrekvens beräknas förstärkningen. Om denna blir lägre än -40 dB avbryts mätningen.

Under mätningens gång skrivs successivt mätresultatet ut i tabellform på skärmen. Då mätningen är avslutad ges valmöjlighet genom en meny att få mätresultatet utskrivet som:

1. En graf på skärmen.
2. Tabellutskrift på printer.
3. Använda printern som plotter för plottning av frekvenskurvan.

FREKV Hz	UTSP VOLT	dB
2000	10.88	1.76
3000	13	3.31
4000	14.87	4.48
5000	16.48	5.37
6000	17.8	6.04
7000	18.77	6.5
8000	19.47	6.82
9000	19.97	7.04
10000	20.3	7.18

STÄLL GENERATORNS OMRÅDESOMKOPPLARE I
LÅGE FREKVENS * 10000
TRYCK SEDAN PÅ RETURN ■

Bild 70a. Mätresultatet presenteras på bildskärmen dekad för dekad.

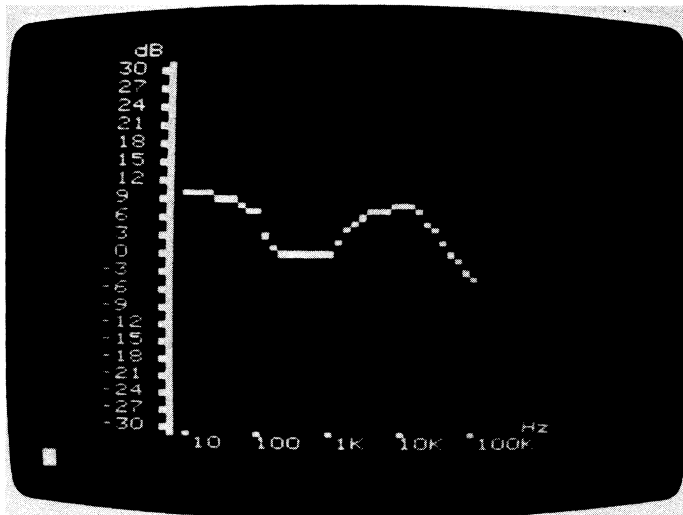


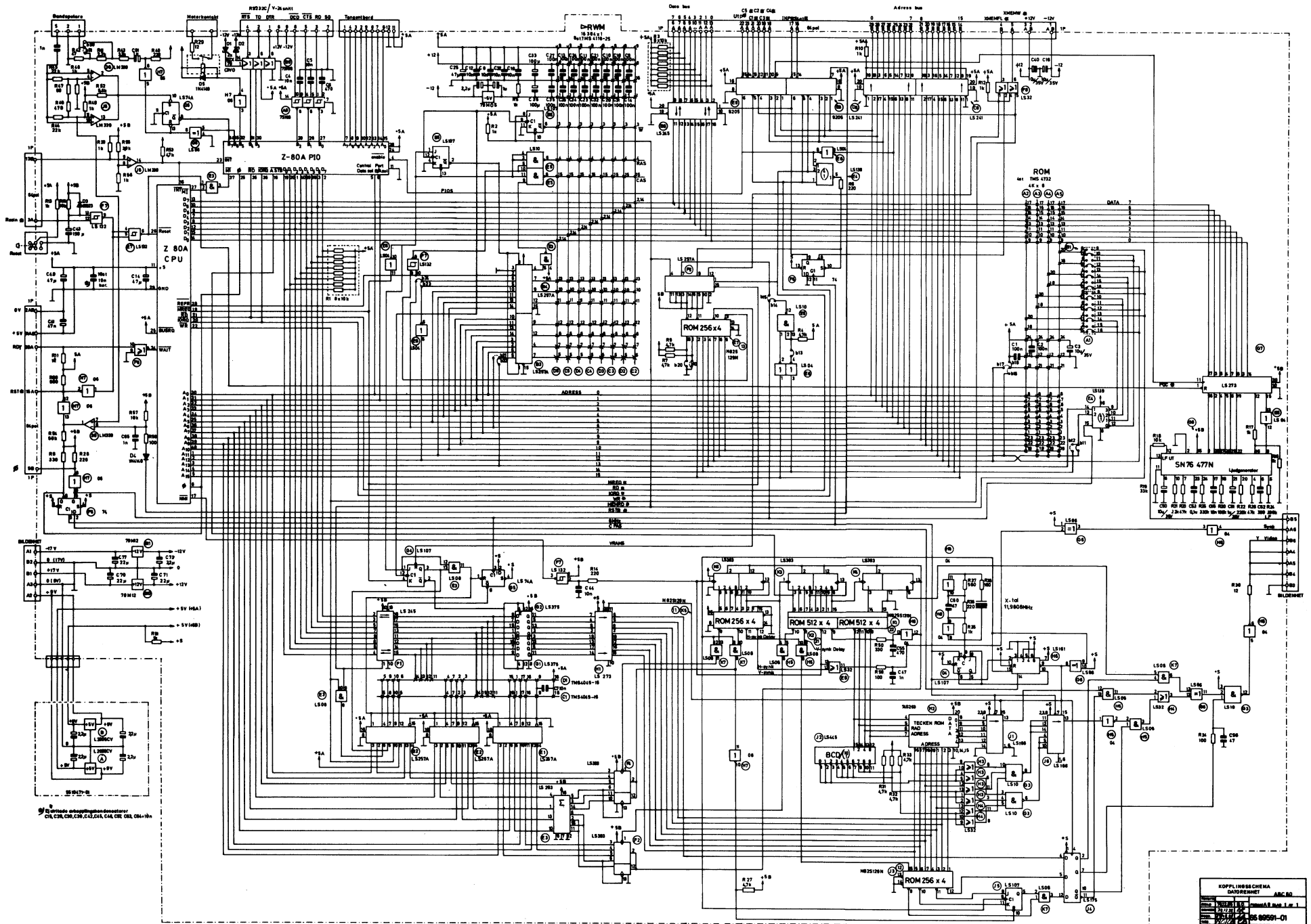
Bild 70b. Efter avslutad mätning kan man begära grafutskrift på bildskärmen.

FREKV HZ	UTSP VOLT	DB
10	.42	-7.81
20	1.81	4.87
30	2.67	8.25
40	3.08	9.49
50	3.28	10.04
60	3.36	10.25
70	3.37	10.27
80	3.34	10.2
90	3.29	10.06
100	3.23	9.91
200	2.51	7.71
300	1.96	5.57
400	1.637	4
500	1.438	2.88
600	1.383	2.82
700	1.297	1.36
800	1.137	.84
900	1.08	.39
1000	1.833	0
2000	.745	-2.84
3000	.568	-5.19
4000	.451	-7.2
5000	.365	-9.83
6000	.306	-10.56
7000	.26	-11.98
8000	.225	-13.24
9000	.196	-14.43
10000	.1723	-15.55
20000	.058	-25.01
30000	.0218	-33.51

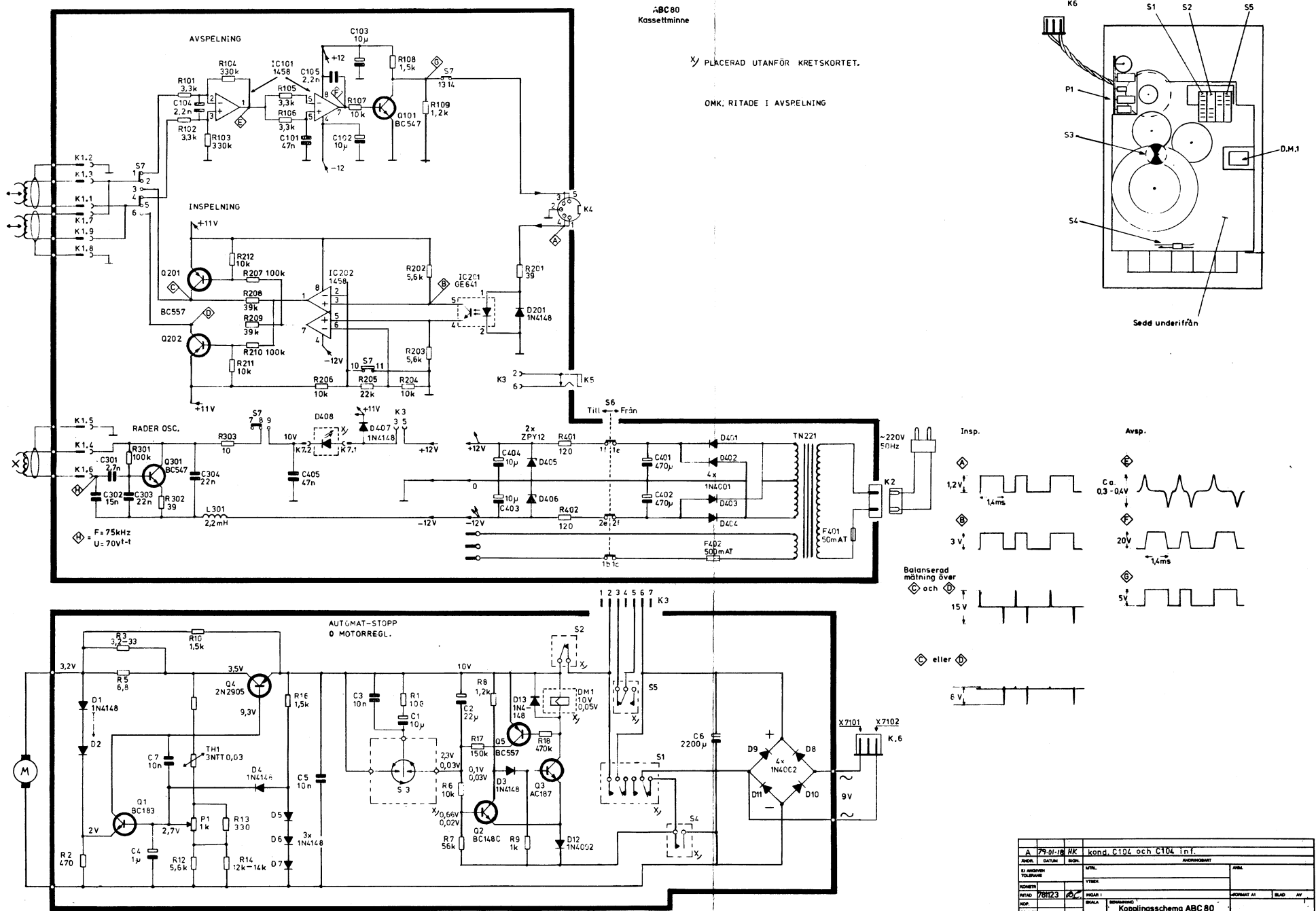
Bild 71a. Ex. på tabellutskrift.

b ₇ b ₆ b ₅		b ₄ b ₃		b ₂ b ₁		b ₀		MSG		MSG		MSG		MSG		MSG																																												
B		b ₄ b ₃		b ₂ b ₁		b ₀		MSG		MSG		MSG		MSG		MSG																																												
row 1		row 1		row 1		row 1		MSG		MSG		MSG		MSG		MSG																																												
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																											
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0																																											
0	0	0	1	1	SOH	DC1	LLO	0	1	1	A	Q	0	1	1	0	0																																											
0	0	1	0	2	STX	DC2	"	0	2	2	B	R	0	1	1	0	0																																											
0	0	1	1	3	ETX	DC3	#	0	3	3	C	S	0	1	1	0	0																																											
0	1	0	0	4	EOT	DC4	\$	0	4	4	D	T	0	1	1	0	0																																											
0	1	0	1	5	ENQ	PPC ^①	%	0	5	5	E	U	0	1	1	0	0																																											
0	1	1	0	6	ACK	SYN	&	0	6	6	F	V	0	1	1	0	0																																											
0	1	1	1	7	BEL	ETB		0	7	7	G	W	0	1	1	0	0																																											
1	0	0	0	8	BS	GET	SPE	0	8	8	H	X	0	1	1	0	0																																											
1	0	0	1	9	HT	TCT	EM	0	9	9	I	Y	0	1	1	0	0																																											
1	0	1	0	10	LF	SUB	*	0	10	10	J	Z	0	1	1	0	0																																											
1	0	1	1	11	VT	ESC	+	0	11	11	K	[0	1	1	0	0																																											
1	1	0	0	12	FF	FS	,	0	12	12	L	\	0	1	1	0	0																																											
1	1	0	1	13	CR	GS	-	0	13	13	M]	0	1	1	0	0																																											
1	1	1	0	14	SO	RS	.	0	14	14	N	^	0	1	1	0	0																																											
1	1	1	1	15	SI	US	/	0	15	15	O	_	0	1	1	0	0																																											
								0	1	1	?	UNL	0	1	1	0	0	0	0																																									
								0	1	1	0	0	1	1	0	0	0	0	0	0																																								
								0	1	1	0	0	0	1	1	0	0	0	0	0																																								
								0	1	1	0	0	0	0	1	1	0	0	0	0																																								
								0	1	1	0	0	0	0	0	1	1	0	0	0																																								
								0	1	1	0	0	0	0	0	0	1	1	0	0																																								
								0	1	1	0	0	0	0	0	0	0	1	1	0	0																																							
								0	1	1	0	0	0	0	0	0	0	0	1	1	0	0																																						
								0	1	1	0	0	0	0	0	0	0	0	0	1	1	0	0																																					
								0	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0																																				
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																																			
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																																		
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																																	
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																																
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																															
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																														
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																													
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																												
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																											
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																										
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																									
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																								
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																							
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																						
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																					
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																				
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																			
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																		
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																	
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0																
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0															
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0														
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0													
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0												
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0											
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0										
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0									
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0								
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0							
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0						
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0					
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0				
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0			
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0			
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0		
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	
								0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
								0	1	1	0	0	0	0																																														

Bilaga 2: Kopplingschema datorenhet



Bilaga 3: Kopplingschema ABC 80 kassetminne



A 79-01-R JK		Kond. C104 och C104 inf.	
PROJ.	DATEM	SKID.	ANORDNINGAR
EJ ANSVARIG FÖLJANDE		ARB.	
MANT.		YTREK.	
REVIS.	7/21/23	PROJ.	SKID.
DRÖJ.		SKALA	ANORDNINGAR
Kopplingschema ABC 80 Kassetminne		FORMAT A1	BLAD 01
DOC.	79-01-R JK	86-89589-01	
IND.	79-01-R JK	LUXOR INDUSTRI-AKTIEBOLAG METALA	

Referenslitteratur

Gunnar Markesjö, *Mikrodatorns ABC*. Esselte Studium.

ABC om mätatorsystem. Scandia Metric AB.

Bygg ut ABC80 med Databoard 4680. SATTCO AB.

ABC om Basic. DIDACT.

Jan Lundgren — Bengt Lundin, *ABC om programmering och dokumentation*.

EMMDATA.

Anders Isaksson — Örjan Kärrsgård, *Avancerad programmering på ABC80*.

Studentlitteratur.

ABC om användardokumentation. LUXOR AB.

Olle Felten, *BASIC-handbok för ABC80*. Studentlitteratur.

Lennart Råde, *Att programmera med ABC80*. Studentlitteratur.

Sakregister

- ABC-bussen 36
- ABC-bussens stift-
konfiguration 37
- ABC80 som funktions-
generator 82
- adressering av I/O-kort 52
- adressval 96
- analog-digitalomvandlare AD 63
- anslutning av snabbskrivare 54
- automatisk mätvärdes-
insamling 86

- basadress 48
- basicdialekt 45
- basictolk 45
- buffrade utgångar 17
- bussens struktur 94
- byglingsplugg 48

- card select (CS) 52
- Centronics-snittet 54
- controller 95
- controllerkort 41
- CS (card select) 36

- DAC 69
- databussen 36
- DIN-kontakt (41612) 36
- Disc Operativ Systemet
(DOS) 42
- DOS-buffert 45
- double density 39

- error handler 27
- expansionsmöjligheter med
ABC80 39

- felhanteringsrutin 27

- flexskiveminne 39
- floppydisc 39
- formfeed 54

- givare 23
- golv 50
- GPIP 91

- HBIB 91

- identifieringsinformation 20
- IEC-bussen 91
- IEC-bussmätning av
frekvensgång 107
- IEEE 488 91
- initiering av printar 54
- inkoppling 58
 - av AD 64
 - av IEC-utrustningen 101
- inläsning från display 15
- INP (58) 11
- interface 54
- interpretator 45
- I/O-enheter 8
- I/O-kommandon för 4007 60
 - för 4082 66
 - för 4083 72

- kassettbuffert 45
- kontrollbussen 36
- kortval 52

- laborationer med reläkort 60
- laborationer på AD 67
- listener 95
- logik monitor 29

- minnesbussen 36

minnesexpansion 48
minnesstacken 45
multiplexer 29
mätning med IEC-buss 103

OUT 58,X 11

piggy back 91
PIO-krets 9
processkontroll 23
programmering av
IEC-bussen 99
programmering av V:24 9

RAM 45
RAM-area 45
ROM-area 45
RS 232 C 8

spänningsutgång 70
ström utgång 70
styrning av display 14
av nio switch-funktioner 17
styrprogram 22
SWAP% 74
system för komponent-
provning 84
systemvoltmeter 103

tak 45
talk only mode 103
talker 95
telemetrisystem 23
tillämpningskopplingar för
V:24-snittet 14

UNL 99

V:24 8

Z=7 AND INP(58) 12

åtta reläutgångar 58

4680-bussen 36



Åke Westh

STYR och MÄT med ABC80

Boken vänder sig i första hand till gymnasieskolans tekniskt inriktade linjer, motsvarande arbetsmarknadsutbildningar samt till industrin. För att kunna tillgodogöra sig bokens innehåll fordras kunskaper i BASIC-programmering, digitalteknik och grundläggande datalära.

Första delen av boken behandlar V:24-snittet. I detta avsnitt behandlas programmering, styrning och mätning. Exempel på detta är in- och utläsning från extern display, styrning av nio buffrade utgångar, processövervakning av åtta mätställen samt användning av ABC80 som Logicmonitor.

De framtagna konstruktionerna i V:24-avsnittet finns inte att tillgå på den öppna marknaden.

Avsnittet om ABC-busstillämpningar behandlar minnesexpansion och utnyttjande av minnesstacken. Ett antal systemtillämpningar baserade på användning av DAC, AD, reläkort och IEC-buss framställs på I/O-sidan. Exempel på detta är svepgenerering, komponentprovning av resistor, automatisk temperaturövervakning och styrning av åtta utgångar samt automatisk uppmätning av frekvensutgångar i en förstärkare.

Mikrodatorer och Digital teknik — ett urval

Danielsson: Digital teknik

Danielsson m fl: Digital teknik — övningsbok

Gralén : Digitala kretsfamiljer

Gudmundsson: Virtuellt minne

Hallberg: Genväg till mikrodatorn

Hallberg: Hur datorer fungerar

Hallberg: Minnen och In/Ut-enheter

Hemert: Grundläggande digitalteknik

Hemert: Grundläggande digitalteknik — Arbetsbok

Hemert: Grundläggande mikrodatorteknik

Hemert/Blomdell: Grundläggande mikrodatorteknik — Arbetsbok

Jennel/Wallin: Datorteknik och programmering

Markesjö: Bli bekant med en minidator

Nelson/Wennersten: Hemdatorrevolutionen

Siemens AB: Mikrodatorer i ett nötskal

Wennersten: Privatdatorn — din egen dator

Wennersten: Vår elektroniska framtid

Älmeby: Mikrodatorn — en introduktion

Älmeby: Konstruera och programmera med 8748



Studentlitteratur – ett förlag
inom Utbildningshuset

Box 1719, 221 01 Lund. Tel. 046-30 70 70

ISBN 91-44-17631-7