

ABC BLADET

ABC-KLUBBENS MEDLEMSBLAD FÖR BLANDAD INFORMATION TILL BÅDE NYTTA OCH NÖJE

NUMMER 1, 1980

ÄNTLIGEN IGÅNG!

Så har då ABC-klubben äntligen kommit igång på allvar!

För den som eventuellt inte känner till oss, är vi alltså en förening för användare av 1 svenska mikrodatoren ABC80.

Vid vårt konstituerande möte den 24:e jan. i år, samlades en ovanligt stor skara entusiaster på Tekniska Högskolan i Stockholm.

Under ordförandeskap av Gunnar Markesjö, bildade de 237 närvarande inte mindre än 15 olika intressegrupper. Grupperna representerar ADB-tillämpningar inom t.ex.: administration, datakommunikation, föreningsliv, matematik, programmeringsteknik, spel & grafik & musik, styr- & mätteknik, textbehandling, etc.

SYFTET MED ABC-KLUBBEN

Enligt interimsstadgarna är syftet att verka för att tillvarata medlemmarnas gemensamma intressen av datorer och dator-tillämpningar, och verka för ökade kunskaper inom dessa områden till nytta, utbildning och nöje."

Det råder ingen tvekan om att ABC-klubben redan nu lever upp till sin formella målsättning!

Entusiasmen och energin inom de olika sektionerna är rentav förbluffande!

Då olika tillämpningar ofta griper in i varann, är många med i flera sektioner samtidigt. Man kan alltså tala om mycket aktiva "tvär-intresserade" medlemmar.

VILKA ÄR MEDLEMMAR?

Alla slags människor inom de mest osannolika yrken använder ABC80.

VEM FÅR BLI MEDLEM?

Du t.ex.!

Det är inte nödvändigt att man måste äga en ABC80 för att bli medlem. Det räcker om man är intresserad av att lära sig mer om hur datorn fungerar, eller är allmänt ADB-intresserad. Välkommen alltså!

John Kvarnström

VÄLKOMNA TILL ÅRSMÖTET

Härmed kallas medlemmarna till årsmöte med ABC-klubben, tisdagen den 22 april, 1980 kl. 19.00

Lokal: Tekniska Högskolan, sal F1, Lindstedtsvägen 22, Stockholm.

Dagordning, förslag till reviderade stadgar samt valberedningens förslag till val av styrelse och revisorer kommer att utdelas vid sammanträdet.

På årsmötet kommer bl.a. demonstration att ske av Gunnar Tidners program för terminalkommunikation mellan klubbens ABC-80 och medlemmarna. Se separat artikel på annat ställe i klubbladet.

Väl mött på årsmötet !

För styrelsen

Odd Rolander
Ordförande

ABC80 Systemvariabler

***** VARIABEL *****

VARIABELLISTAN I ABC80 HAR EN LÄNKAD LISTA, DRR VARJE VARIABEL I LISTAN PEKAR UT NÄSTA VARIABEL.

LISTAN BÖRJAR PÅ ADRESS 65065-65066.

FÖR VARJE VARIABEL FINNS ETT ANTAL BYTES SOM NÄRHARE BESKRIVS NEDAN.

```

HELTAL          N B P P V V
HELTALSVEKTOR  N B P P A A M M T T
HELTALSMATRIS  N B P P A A M M T T X X Y Y
FLYTAL         N B P P V V V V V V
FLYTALSVEKTOR  N B P P A A M M T T
FLYTALSMATRIS  N B P P A A M M T T X X Y Y
STRÄNG         N B P P O D R R L L
STRÄNGVEKTOR   N B P P O D M M T T
STRÄNGMATRIS   N B P P O D M M T T X X Y Y
    
```

```

N      RR TYP OCH NUMMERBETECKNING
      BITARNA TOLKAS ENLIGT FÖLJANDE.
      NNNN0001 HELTAL
      NNNN0101 HELTALSVEKTOR
      NNNN1001 HELTALSMATRIS
      NNNN0000 FLYTAL
      NNNN1000 FLYTALSVEKTOR
      NNNN1000 FLYTALSMATRIS
      NNNN0110 STRÄNG
      NNNN0110 STRÄNGVEKTOR
      NNNN1010 STRÄNGMATRIS
      NNNN RR VARIABELNS NUMMERBETECKNING DRR 0000 STAR FÖR
      T.EX. AD 10001 FÖR A1, 1111 STAR FÖR BARA A.
      RR VARIABELNS BOKSTAVSBETECKNING.

P P    RR EN PEKARE SOM PEKAR UT NÄSTA VARIABEL
V V    RR VARIABELNS VÄRDE.
A A    RR ADRESSEN TILL VARIABLERNA. VARIABLERNA LIGGER SÅ ATT
      DET HÖGRA INDEXET VARIERAR SNABBAST.
M M    RR DET TOTALA ANTALET VARIABLER.
X X    RR DEN ÖVRE GRÄNSEN FÖR VÄNSTRA INDEXET.
Y Y    RR DEN ÖVRE GRÄNSEN FÖR DET HÖGRA INDEXET.
D D    DIMENSIONERAD LÄNGD FÖR STRÄNGEN.
R R    PEKARE TILL STRÄNGEN.
L L    AKTUELL LÄNGD PÅ STRÄNGEN.
O O    PEKARE TILL TABELL ÖVER STRÄNGBESKRIVNINGAR.
      D D R R L L D D R R L L ...
      BESKRIVNINGARNA LIGGER SÅ ATT DET HÖGRA INDEXET VARIERAR
      SNABBAST.
T T    ANVÄRNS AV ABC80S BASIC FÖR TILLFÄLLIG LAGRING UNDER BERÄKNING.

HELTAL LIGGER LAGRADA SÅ ATT DEN MIST SIGNIFIKANTA BYTEN KOMMER FÖRST.
FLYTAL BESTÅR AV 5 BYTES SSSTE, DRR SSS RR HANTISSAN MED TVA BCD-
SIFFROR I VARJE BYTE. T RR TALET'S TECKEN. E RR 40-EXPONENTEN EXCESS 128.
    
```

LÄT OSS SOM EXEMPEL TA NEDANSÄNDE PROGRAM:

```

10 AX=25
20 DIM B$(12)=3X
30 B$(02)="B0"
40 B$(12)="B-1"
50 DIM C(12,12)
60 C(02,02)=200
70 C(02,12)=201
80 C(12,02)=210
90 C(12,12)=211
100 STOP
    
```

PROGRAMMET GER UPPHOV TILL FÖLJANDE VARIABELLISTA, FÖRST VISAS ADRESSEN DECIMALT, SEDAN INNEHALLET I MINNESPOSITIONEN DECIMALT, HEXADECIMALT, ASCII OCH SIST BINÄRT. PEKAREN PÅ ADRESS 65065-65066 PEKAR PÅ ADRESS 49346.

```

49346 241 F4 11110001 & NAMN AX
49347 65 41 A 01000001 &
49348 200 C8 11001000 > 49352
49349 192 C0 11000000 &
      350 25 49 00010001 & VRDEY 25
49351 0 00 00000000 &
-----
49352 246 F6 11110110 & NAMN B$( )
49353 66 42 B 01000010 &
49354 210 D2 11010010 > 49362
49355 192 C0 11000000 &
49356 224 E0 11100000 >> 49376
49357 192 C0 11000000 >>
49358 1 01 00000001 & MAX V
49359 0 00 00000000 &
49360 6 06 00000110 &
49361 0 00 00000000 &
-----
49362 248 F8 11111000 & NAMN C( )
49363 67 43 C 01000011 &
49364 0 00 00000000 > SLUT
      365 0 00 00000000 >
49366 242 F2 11110110 >> 49394
49367 192 C0 11000000 >>
49368 3 03 00000011 & MAX U
49369 0 00 00000000 &
      370 5 05 00000101 &
49371 4 04 00000100 &
49372 1 01 00000001 & MAX X
49373 0 00 00000000 &
49374 1 01 00000001 & MAX Y
49375 0 00 00000000 &
-----
49376 3 03 00000011 & MAXL 3
49377 0 00 00000000 &
49378 236 EC 11101100 > 49380
49379 192 C0 11000000 >
49380 2 02 00000010 & LRNGD 2
49381 0 00 00000000 &
-----
49382 3 03 00000011 & MAXL 3
49383 0 00 00000000 &
49384 239 EF 11101111 > 49391
49385 192 C0 11000000 >
49386 3 03 00000011 & LRNGD 3
49387 0 00 00000000 &
-----
49388 66 42 B 01000010 B$(0)
49389 48 30 0 00110000
49390 0 00 00000000
-----
49391 66 42 B 01000010 B$(1)
49392 45 2D - 00101101
49393 49 31 1 00110001
-----
49394 32 20 00100000 C(0,0)
49395 0 00 00000000
-----
49396 0 00 00000000
49397 0 00 00000000
49398 131 83 10000011
-----
49399 32 20 00100000 C(0,1)
49400 16 10 00010000
49401 0 00 00000000
49402 0 00 00000000
49403 131 83 10000011
-----
49404 33 21 00100001 C(1,0)
49405 0 00 00000000
49406 0 00 00000000
49407 0 00 00000000
49408 131 83 10000011
-----
49409 33 21 00100001 C(1,1)
49410 16 10 00010000
49411 0 00 00000000
49412 0 00 00000000
49413 131 83 10000011
    
```

ABC80 SYSTEMVARIABLER 791116 -1-

HEX	DEC	ANTAL BYTE	ANVÄRNING
FDF0	65008	3	KLOCKAN
FDF3	65011	1	NUVÄRANDE MARKÖRPOSITION PÅ SKRÄMMEN
FDF4	65012	1	..RAD
FDF5	65013	1	BIT 7 SATT ANGER ATT EN TANGENT TRYCKTS NER
FE07	65021	1	BIT 7 SATT ANGER ATT CTRL-C TRYCKTS NER
FE0A	65034	1	BÖRJAN PÅ LISTA ÖVER TILL DEVICE
FE1C	65052	2	BOFA
FE1E	65054	2	EOFA
FE20	65056	2	HEAP
FE27	65063	2	STACKPEKARENS STARTADRESS
FE2B	65067	1	SENASTE SELECTAT KORT MED OUT 1,N
FE32	65074	2	PEKARE TILL LISTA ÖVER ÖPPNA FILER

Har du svårt att hålla reda på vilka variabler du använt i ett program?

Ett nyttigt exempel på hur systemvariablerna till vänster kan användas, är följande korta men förträffliga program. Programmet kan lista ut en förteckning av de variabler som förekommer i ett program. Det listar t.o.m indexerade variabler.

Skriv alltså in programmet, och glöm inte några semikolon, samt se till att antalet mellanslag (blanka) överensstämmer. Spara programmet med SAVE VARIABEL, tex. Kör sedan LOAD för det program som du vill ha variablerna listade för. Därefter kör du MERGE VARIABEL, och RUN. Nu får du variablerna listade på skärmen en och en, genom att trycka på godtycklig tangent.

Du som har en printer, ser säkert hur man lätt kan modifiera PRINT-satserna så att man får utskriften på papper.

Tänk på att listprogrammet även listar sina egna variabler (Å% samt Ö%), samt att det andra programmet du vill ha variablerna listade för, inte får börja med radnummer 1, eller ha radnumren 65000 - 65011.

H.N / J.K.

1 GOTO 65000

65000 Å%=65063%

65001 Å%=PEEK(Å%+2)+SWAP%(PEEK(Å%+3%))

65002 IF Å%=0% END

65003 IF PEEK(Å%+1%)=94% GOTO 65001

65004 ; CHR\$(PEEK(Å%+1%));

65005 ; MID\$("0123456789",PEEK(Å%)/16%+1%,1%);

65007 ; MID\$(" %\$",PEEK(Å%) AND 3%)+1%,1%);

65008 ; MID\$(" () ",(PEEK(Å%) AND 12%)+1%,4%);

65009 GET Ö% : REM successiv listning på skärmen.

65010 GOTO 65001

65011 REM OBS att Å% och Ö% ovan också listas.

DATAKOMMUNIKATION

Kommunikationsgruppen hade sitt första möte tisdagen den 4 mars. Vi tittade på några olika användningar av ABC-80 för datakommunikation. Gunnar Tidner visade sitt system ABC-Monitor, en uppbringbar ABC-80 som kan sända och ta emot filer från en annan ABC-80.

Efter att ha sett ABC-80 som terminal till DataVisionen och MetricVisionen, fick vi också demonstrerat hur ABC-80 kan användas som "linjelyssnare" på datakommunikationslinjer.

Gruppen diskuterade också hur man i klubbens regi skulle kunna sätta upp ett ABC-80 system för kommunikation mellan medlemmarna. Leverantörerna kan nog fås att bidra med utrustningen, problemet är att hitta en lämplig plats att ställa upp utrustningen på.

Hans Nilsson

EXTRACT

PLOCKA UT EN DEL AV DITT PROGRAM.

Här kommer ett intressant och användbart program, med vars hjälp man kan ta ut en del av ett annat (eget) program, och spara delen som en separat programfil. Nyttigt vid modul-programmering.

Programmet, som är gjort av Gunnar Tidner, är självinstruerande. Prova med ett testprogram, så lär du dig snart funktionen.

```
GOTO 5000 : REM av Gunnar Tidner efter ide' av Johan Finnved
5000 REM start EXTRACT-program
5010 ; " **** EXTRACT-program ****"
5020 ; "Vill du ha instruktioner?"; : GET A$ : ; A$
5030 IF (ASC(A$) AND 223%)=74 THEN GOSUB 5100
5040 ; "Radnr 1:a rad som skall vara kvar:" : INPUT N%
5050 ; "Radnr 1:a rad i slutsegmentet(tas bort):" : INPUT M%
5060 POKE CALL(3897%,M%),1
5070 A%=CALL(3897%,N%)
5080 POKE -484%,A%,SWAP%(A%)
5090 END
5100 ; CHR$(12);"Instruktioner:"
5110 ; ; "Programmet tar bort rader i början resp"
5120 ; "i slutet av annat program"
5130 ; ; "Gör först LOAD av det program från"
5140 ; "från vilket du vill ta bort rader"
5150 ; "Gör sedan MERGE EXTRACT"
5160 ; "-vilket utnyttjar radnr över 5000"
5170 ; "samt radnr 1"
5180 ; "Kör programmet med RUN"
5190 ; "Gör sedan SAVE program-namn"
5200 ; "Tryck därefter in RESET-knappen"
5210 ; "För att återställa omflyttade pekare"
5220 ; ; "Tryck tangent när du är klar!" : GET C$
5230 RETURN
```

*** TERMINAL ***

För ABC-80 användare.

Ett exklusivt kommunikationspaket utvecklat av v ordf i ABC-klubben ställs här till klubbmedlemmarnas förfogande.

Med hjälp av terminalrutinen T80-PRT och detta kommunikationsprogram ABCMINI kan klubbmedlemmar hos klubbens dator hämta hem både program och textfiler. Det fullständiga kommunikationspaketet kommer att marknadsföras av bl.a Scandia-Metric AB i Solna. Då får Du möjlighet att överföra program och textfiler (båda lagrade i ASCII-format) i båda riktningarna. Du kan med programmen även kommunicera med stordator t.ex DEC-10.

Här kommer ABCMINI. Hur programmet används kommer att demonstreras på ABC-klubbens årsmöte. Tills vidare alltså ingen ytterligare information än att modem för 300 bit/s (överföring av 30 tecken/sekund) skall användas mot klubbens dator. O.R.

```
10 REM ABCMINI filöverföring till egen ABC-80
11 REM Program av Gunnar Tidner
20 OPEN "PR:KB.1" ASFILE 1 : L%=1
30 Z$="/*" : REM Filslutmärke
40 DIM B$=119
50 ; " *** Half Duplex ****"
60 INPUT #L%,A$
70 ; "Överföring av fil från monitorn?"
80 ; "Bekräfta:"; : GET C$ : ; C$
90 ; "Filnamn i MONITORN:"; : INPUT F1$
100 ; "Eget filnamn:"; : INPUT F2$
110 ; #L%,"Hejsan"
120 INPUT #L%,B$
130 IF B$<>"SYNK" THEN 110
140 FOR T=1 TO 100 : NEXT T
150 ; #L%,"SYNKSVAR"
160 ; #L%,F1$
170 INPUTLINE #L%,B$ : B$=LEFT$(B$,LEN(B$)-2%)
180 IF B$<>"OPENED" THEN ; B$ : FOR T=1 TO 3000:
NEXT T : GOTO 60
190 PREPARE F2$ ASFILE 2
200 FOR I%=1 TO 2000
210 ; #L%,'>'
220 INPUTLINE #L%,B$
230 IF LEFT$(B$,2%)=Z$ THEN 270
240 ; #2,B$;
250 ; I%
260 NEXT I%
270 CLOSE 2
280 ; CHR$(7)"Överföringen klar!"
290 ; "Filnamn: ";F2$; ", antal rader=";I%-1%
300 FOR T=1 TO 2000 : NEXT T
310 ; #L%,"Tack!"
320 GOTO 60
```

Insändare

MINNESBRIST

Jag tycker att bildandet av ABC-klubben är enormt positivt. Alla vi "nybörjare" som har huvudet fullt med ideer, vilka vi på grund av kunskapsluckor har svårt att snabbt realisera, kanske äntligen får en chans.

Tekniken överhuvudtaget är ju idag så fördjupad i sektorer att vi bara kan ha specialkunskaper inom ett fåtal av dessa. Mikroprocessorn infiltreras, som alla vet, precis överallt. Det är oerhört spännande, men samtidigt skrämmande. Bland alla positiva aspekter på bildandet av ABC-klubben är det speciellt två som jag för närvarande känner mest för. Den ena är att människor med kunskaper inom skilda områden kommer tillsammans för att kläcka ideer och lösa problem. Ofta sitter var och en för sig och ser inte skogen för bara träd. Problemet man brottas med kanske redan är löst inom en annan teknikgren. Vi behöver i Sverige idag vitamininjektioner av det här slaget för att ta oss upp ur säkerhetsnätet som vi ligger och väntar på bättre tider i och konkurrera med utländsk innovation. Den andra är den behovs- och nivåanpassade utbildning som kommer att växa fram inom ABC-klubbens hank och stör. Skulle vi, som inte är direkt purunga och ej haft teleteknik som huvudämne, lära oss allt om digitalteknik, alla programspråk, osv osv, så skulle våra ideer ej komma oss själva eller andra tillgodo innan vi går i graven. Dock måste vi ju lära oss så mycket som det någonsin är möjligt om den här tekniken för att överhuvudtaget veta vilken hjälp vi behöver få från andra. Även alla andra typer av specialkunskaper måste vara värdefulla bidrag i en "tvär-utbildning", som det här kan bli fråga om.

Till sist skall jag bara säga vad jag tycker om lokalavdelningar. "Explosion" tror jag. Skulle inte ABC-klubben kunna bilda en ekonomisk förening, köpa in maskiner och hyra ut till lokalavdelningarna? Då tror jag att dem kommer snabbare. Intresset finns, var så säker. Med hjälp av terminalkommunikation över hela landet kan ständiga konferenser pågå, oj, oj vad detta skulle kunna ge av nyttig lärdom. Tanken svindlar, tänk om man kunde selektera just de informationsbitar man är intresserad av och lagra på disk.

Christer Bolt

När man tar fram program, händer det att man plötsligt råkar ut för ERR 3, det vill säga maskinen påstår att det inte finns minne så det räcker till. Detta händer alltid när man inte har tillgång till inköpsställen som saluför extra minne. Eller så har man redan så mycket minne det går att stoppa i ABC 80. Vad gör man?

Det första man gör, är att kontrollera att det inte beror på en felaktig instruktion. PRINT SPACE\$(en variabel) ger till exempel samma fel om variabeln råkar vara negativ. Programmet går igenom, alla kommentarer och onödiga variabler tas bort. Det hjälper ändå inte. Ni kan vara lugna, det finns en möjlighet till. Med flexskiveenhet, genererar AEC 80 åtta DOS-buffrar, vardera om 256 tecken. Så många behövs nästan aldrig. Dessa kan reduceras till antalet med hjälp av POKE 65064, värde. Värdet multiplicerat med 256 ger adressen till första DOS-buffer, man väljer alltså själv hur många man vill ha kvar. Om värdet sätts till 251, blir adressen till första buffern 64256, sex av de åtta DOS-buffrarna är borttagna och två finns kvar. Det klarar man sig med, för det mesta. Användarminnet ökar med cirka 1,5 K, i detta fall. POKE-instruktionen kan även användas i program.

GÖ

Omvänd video

* OMVÄND ABC-80 *

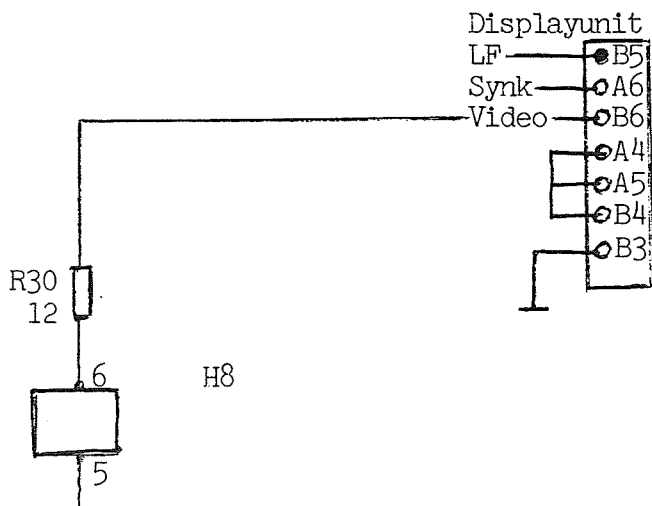
Möjlighet finns för Dig som är lite händig och ej är räddhågsen att * OMVÄNDA * Din ABC-80. D.v.s. att kunna få även omvänd video på bildskärmen (svart text på vit botten). Detta kan utföras av den som är bekant med lödkolven med följande enkla koppling. De enda komponenter som behövs är en tvåpolig tvåvägs miniatyrströmbrytare med on-off-on-funktion, ett motstånd på 220 ohm samt kopplings-sladd.

Observera att ingrepp som Du gör i tangentbordet medför att tillverkarens garanti kan upphöra att gälla!

Kopplingen har använts under mer än ett års tid utan att några fel visat sig uppkomma. Omkoppling sker enkelt på baksidan av tangentbordet mellan normal och omvänd video. Du får då även möjlighet att i brytarens mittläge släcka bildskärmen.

Fördelarna med att kunna köra datorn med omvänd video är stora genom att reflexer i bildskärmen bättre elimineras. Det är mindre tröttande att sitta framför bildskärmen om det är ljust i rummet och det kan vara trevligt att kunna presentera grafik även med omvänd video.

Kopplingsschemat i de delar som berörs ser ut så här:

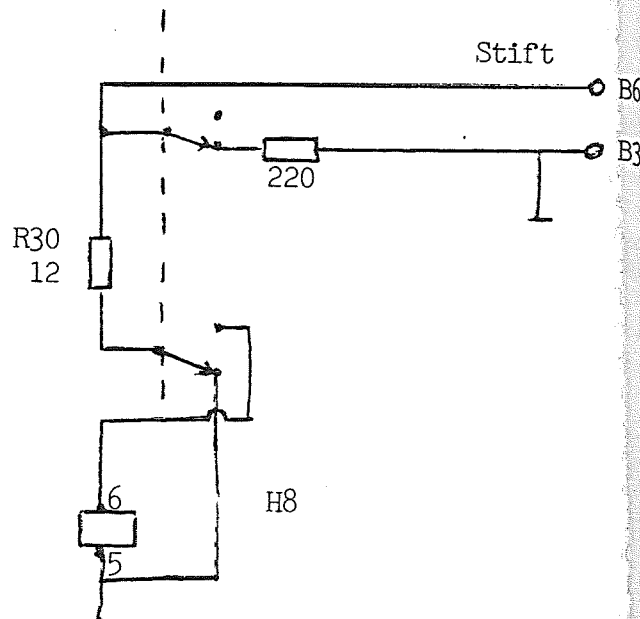


Det enda ingrepp som behöver göras är videosignalen tas ut före stift 5 på IC-kretsen H8 och leds till jord via ett 220 ohms motstånd.

Anslutning sker till stift B3 på kontakten till bildskärmsenheten. Samtidigt skall videosignalen gå till stift B6 i kontakten.

Kopplingen kan enkelt göras enligt följande.

All lödning bör göras med en lödkolv med låg effekt, 20 - 30 watt. Lödkolven bör ha jordad lödspets. Tänk på att inte vara statiskt uppladdad då Du arbetar. Arbeta utan skor på fötterna för att avleda statisk uppladdning. Lägg då och då handen mot något metallföremål för att avleda statisk elektricitet. Elementet rekommenderas inte om Du samtidigt håller i lödkolven eller på annat sätt är förbunden med 220 volt.



Ta loss kåpan över tangentbordet, tangentbord, kylfläns samt ta ut datorkortet så att lådan blir tom. Observera att datorkortet både är monterat med skruvar uppifrån och med snäpplås från undersidan. Borra hål i chassilådan för strömbrytaren mellan utgångshålet för motorstyrning och hålet för kontakten till bildskärmsenheten. Använd en brytare av miniatyr-typ och borra hålet så högt upp i chassilådan att kontakten får plats ovanför kretskortet när detta åter skall sättas på plats. Se till att inga borrar-spån blir kvar i lådan som kan komma att kortsluta kretsar. Löd kopplingstrådar på brytarkontakten och dra fram dem till de aktuella lödställena.

Inkoppling före motståndet R30 görs enklast genom att den sida av motståndet som är anslutet till ben 6 på IC-kretsen H8 löds loss från kretskortet och lyfts upp. Anslutning sker sedan till detta motstånd och till det hål i kretskortet som nu blivit tomt. Inkoppling till ben 5 på IC-kretsen H8 görs lättast på undersidan av kretskortet om Du vill undvika att löda direkt på benet på IC-kretsen.

Inkopplingen i övrigt är okomplicerad.

51-

DAGSTART

med
ör

Här kommer ett par hjälpprogram som har visat sig vara väldigt tidsbesparande och användbara. Förbättra dom gärna, och dela med er.

och

nen-
gt
tt

Det första programmet, DAGSTART, frågar efter dagens datum och hur mycket klockan är. Efter beräkning läggs dessa värden in på adress 65008-65010, där ABC 80 håller reda på en intern klocka. Dagens datum har jag lagt i några av bildminnets lediga adresser, 31866 till 31871. Där kommer datumet att ligga orört tills strömmen slås av. Det spelar alltså ingen roll om man trycker på RESET-knappen eller hur man skriver till bildminnet, utom om man använder POKE, så påverkas inte datumet.

B6

B3

Därifrån hämtas sen dagens datum, vid behov, med hjälp av PEEK, se exempel i program LISTA på rad 90 till 130.
GÖ

```

240 Z1=INT(Z/256) : Z1%= NOT Z1
250 Z%=( NOT (Z-Z1*256)) OR 4%
260 POKE 65008%,Z%,Z1%,SWAP%(Z1%)
270 CHAIN "LIB"
280 FOR I%=0% TO 5%
290 GET A$(I%) : IF ASC(A$(I%))=8% I%=
I%-1% : IF I%<0% I%=0% : GOTO 290
ELSE ; A$(I%) ; : GOTO 290 ELSE 300
300 IF ASC(A$(I%))>57% OR ASC(A$(I%))<
48% 290 ELSE ; A$(I%) ;
310 NEXT I%
320 RETURN
    
```

SLA TELEFONNUMMER MED ABC-80!

Om man kopplar in reläet för kassettbandspelaren i serie med telefonapparaten kan man med nedanstående program låta ABC-80 koppla upp ett telefonsamtal. Nedkoppling sker om man skriver RUN och sedan svarar med enbart RETURN. OBS! Endast för interna telefonnät. Får ej anslutas till televerkets nät! OBS!

```

10 OUT 58,0;
20 ; "Vilket nummer önskas"; : INPUT N$
30 GOSUB 60 : END
60 REM SUBROUTIN FÖR NUMMERSLAGNING
70 L%=LEN(N$)
80 OUT 58,32 : REM Koppla in linjen
85 REM Vänta på kopplingston
90 FOR I=0 TO 2000 : NEXT I
100 FOR N%=1% TO L%
110 S$=MID$(N$,N%,1%) : REM Nästa siffra
120 IF ASC(S$)<48 THEN 240
130 IF ASC(S$)>58 THEN 240
131 REM Ej siffra, hopp till vänta.
132 REM Används vid internväxel, t.ex.
133 REM 0+ för att få linjen.
140 S%=VAL(S$)
145 REM Paus mellan siffror:
150 FOR I=0 TO 300 : NEXT I
160 FOR P%=0% TO S%
165 REM Väntetid för avbrott på linjen:
170 FOR I=0 TO 70 : NEXT I
180 OUT 58,0
190 FOR I=0 TO 70 : NEXT I
200 OUT 58,32
210 NEXT P%
220 NEXT N%
230 RETURN
240 FOR I=0 TO 5000 : NEXT I : GOTO 220
    
```

Göran Sundqvist

PROGRAMLISTA DAGSTART.BAS

an-
kor-

ra att
truvar
ersid-
öm-
oto
ill
re av
gt upp
ats

sk

borr-

ma att

ådar

m till

130

140

150

och

160

170

180

190

200

210

220

230

erad.

```

10 REM 790817/790820 PGM DAGSTART
20 ; CHR$(12%)
30 ; "SÄTT DAGENS DATUM (ÅÅMMDD) :";
40 GOSUB 280
50 ; CUR(1%,28%)A$(0%);A$(1%) "-"A$(2%)
;A$(3%) "-"A$(4%);A$(5%)
60 ;
70 ; "ÄR DATUM OK? (J/N)"TAB(27%);":"
;
80 GET B1$ : ; B1$
90 IF B1$<>"J" THEN 20
100 FOR I%=0% TO 5%
110 POKE 31866%+I%,ASC(A$(I%))
120 NEXT I%
130 ; CUR(10%,0%) "SÄTT TIDEN (HHMMSS)"
TAB(27%) ":" ;
140 GOSUB 280
150 ; CUR(10%,28%)A$(0%);A$(1%) "."A$(2%)
;A$(3%) "."A$(4%);A$(5%)
160 ; ;
170 ; "TID OK? (J/N) :";TAB(27%) ":" ;
180 GET B1$ : ; B1$
190 IF B1$<>"J" THEN 130
200 H%=(VAL(A$(0%))*10%)+VAL(A$(1%))
210 M%=(VAL(A$(2%))*10%)+VAL(A$(3%))
220 S%=(VAL(A$(4%))*10%)+VAL(A$(5%))
230 Z=1.8E+5*H%+3000*M%+50*S%
    
```


Programmet kan användas för att lista ut program lagrade på flexskiva med kommandot LIST (program lagrade i basformat). Programmet kan användas för att lista program med olika teckenbredd på printer och automatiskt växla sida beroende på antal rader på papperet.

Programmet, som det här är utformat listar program med 40 teckens bredd på printer där papperslängden är 70 rader.

Radlängden på papperet kan ändras på 340 i programmet. Antalet utskrivna programrader per sida innan sidväxling sker kan ändras på rad 380.

Med programmet får man sina programlistor utskrivna så att hållslagning kan ske i vänsterkanten och listningen behöver inte gå över papperets perforerade skarvar.

Rader som är längre än 40 tecken delas upp på upp till 4 rader. Samtliga programlistningar i tidningen har gjorts med hjälp av programmet.

Programmet bör användas av de klubbmedlemmar som sänder in bidrag till tidningen.

Bredden på den utskrivna programlistan kan ändras till t.ex 80 tecken genom att programraderna 211,212,291,421 och 4 tas bort. På rad 310 tas X5\$ bort och värdet ändras till 54% resp. 55% på rad 180 och 190 samt på raderna 200 och 210.

PROGRAMLISTA LIST40.BAS

```

10 REM 800317/800520 PGM LIST40
20 DIM Z%=200%,Z1%=1%,X1%=6%,X2%=200%
   ,X3%=200%,X4%=60% : F%=1%
30 ; CHR$(12%)CUR(5%,0%);VILKEN FIL
   >^; : INPUTLINE I$ : I$=LEFT$(I$,L
   EN(I$)-2%)
40 Z%=INSTR(1%,I$,'.') : IF Z%=0% I$=
   I$+'.BAS'
50 OPEN ^PR:^ ASFILE F%
60 ; #F% : ; #F%,TAB(4%)^PROGRAMLISTA
   ^I$
70 ; #F%CHR$(10%) : F1%=3%
80 OPEN I$ ASFILE 2%

```

```

90 ONERRORGOTO 430
100 INPUTLINE #2%,Z%
110 Z%=LEFT$(Z$,LEN(Z$)-2%)
120 Z%=INSTR(1%,Z$,CHR$(9%))
130 IF Z%=0% 150 ELSE I%=ASC(RIGHT$(Z$
   ,Z%+1%))
140 Z%=LEFT$(Z$,Z%-1%)+SPACE$(I%)+RIGH
   T$(Z$,Z%+2%) : GOTO 120.
150 Z%=INSTR(1%,Z$,' ')
160 X1%=LEFT$(Z$,Z%-1%)
170 X2%=RIGHT$(Z$,Z%+1%)
180 IF LEN(X2$)>34% 190 ELSE 280
190 X3%=RIGHT$(X2$,35%) : X2%=LEFT$(X2
   $,34%)
200 IF LEN(X3$)>34% 210 ELSE 280
210 X4%=RIGHT$(X3$,35%) : X3%=LEFT$(X3
   $,34%)
211 IF LEN(X4$)>34% 212 ELSE 280
212 X5%=RIGHT$(X4$,35%) : X4%=LEFT$(X4
   $,34%)
220 GOTO 280
230 RESTORE : ONERRORGOTO 270
240 READ Ö$
250 Z%=INSTR(60%,X2$,Ö$) : IF Z%=0% OR
   Z%>68% 240
260 X3%=RIGHT$(X2$,Z%) : X2%=LEFT$(X2$
   ,Z%-1%) : GOTO 280
270 X3%=RIGHT$(X2$,69%) : X2%=LEFT$(X2
   $,68%)
280 GOSUB 380
290 IF X3$<>' ' GOSUB 400
291 IF X4$<>' ' GOSUB 421
310 X5$=' ' : X4$=' ' : X3$=' ' : X2$=' '
320 GOTO 90
330 DATA : ,;,' ',=,<,>,+,-,','
340 FOR F2%=F1%+1% TO 70%
350 ; #F%
360 NEXT F2% : F1%=0%
370 ; #F% : F1%=F1%+1% : RETURN
380 IF F1%>65% GOSUB 340
390 ; #F%TAB(4%)X1$TAB(10%)X2$ : F1%=F
   1%+1% : RETURN
400 ; #F%TAB(10%)X3$ : F1%=F1%+1%
410 IF X4$<>' ' ; #F%TAB(10%)X4$ : F1%=
   F1%+1% : RETURN
420 RETURN
421 IF X5$<>' ' ; #F%TAB(10%)X5$ : F1%=
   F1%+1%
430 ; #F%"" : ; #F%," * READY"
440 END

```

* READY

* ANVÄNDARRAPPORT *
EXPAND-ABC

Det finns en relativt billig och bra möjlighet att utöka minneskapaciteten på ABC-80 med ytterligare 16 kbyte. Man får då 16 "friska" kbyte.

Av de 16 kbyte som sitter i tangentbordet kan endast ca 12,4 k utnyttjas för basic-programmering om man nämligen har printerrutin lagrad i botten och använder alla dosbuffrarna för flexskivehantering. Minnet kan i vissa fall ökas genom att man "pokar bort" dosbuffrar man inte behöver använda. Se om detta under programmeringstips på annat ställe i tidningen.

Genom att installera Expand-ABC i tangentbordet kan man utnyttja datorns hela minneskapacitet. Man får då flexskiveenheten ledig för att installera andra kort i den. Expand-ABC har använts under några månaders tid utan att något fel kunnat iakttas.

Observera att tillverkarens garanti för datorn kan upphöra att gälla om ingrepp görs i tangentbordet!

Expand-ABC består av två små kretskort som trycks ned i de IC-hållare där vissa minneskretsar (RAM-kapslar) nu sitter i tangentbordet. Minneskretsarna flyttas till översidan av de två små kretskorten. Sju st ledare skall därutöver anslutas till kopplingspunkter på kretskortet i tangentbordet.

Priset inkl. montering och moms är ca 2 230 kr. I priset ingår då att det dataföretag som gör installationen även övertar garantin för tangentbordet. Ett års garanti lämnas på Expand-ABC.

Den som vill göra installationen själv kan på vissa håll köpa enbart Expand-ABC för ca 1 870 kr inkl. moms.

Installationen är enkel med några undantag. Har man otur kan IC-hållarna vara så trånga att kontaktstiften till Expand-ABC ej går att pressa ned helt, vilket är nödvändigt för att kontakt skall uppnås. Man kan således tvingas löda loss IC-hållarna och byta ut dem. För att detta arbete skall gå bra att

utföra behövs en tennsug eller kopparstrumpa för att suga upp det gamla lödtennet från kretskortet innan hållarna kan avlägsnas. All lödning bör ske med lödkolv med jordad lödspets och försiktighet iakttas med statisk elektricitet. Se artikel rörande inkoppling för omvänd video på annan plats i tidningen.

Det man även måste se upp med vid installationen är att vissa av de IC-kretsar anslutning skall ske till har 16 ben och andra 14 ben. Man bör alltså räkna rätt på kretsarna när anslutning skall ske.

Har man installerat Expand-ABC går inte de gamla RAM-kapslarna utan vidare att sätta tillbaka i IC-hållarna på tangentbordets kretskort om man skulle vilja avlägsna Expand-ABC. Expand-ABC vidgar nämligen kontakterna i IC-hållarna något när den installeras.

Problemen med installationen skall inte överdrivas, men den som inte har vana vid lödnings- och monteringsarbete av detta slag, bör nog inte försöka att installera kretskorten själv.

Funktionen, när kretsarna väl kommit på plats verkar bra. I installationsbeskrivningen lämnas även anvisning om hur minnestestprogrammet skall ändras för att testa de nyinstallerade kretsarna.

Anvisningen medför att inte hela RAM-minnet testas eftersom programmet ligger i botten. Detta sker för övrigt ej heller med programmet när det används med endast 16 k minne. Med 32 k minne kan testprogrammet även laddas in på adress 49152 genom att först ge kommandot POKE 65052,0,192 varefter test kan ske av adresserna 32768 - 49151 genom att lägga in dessa adresser i testprogrammet.

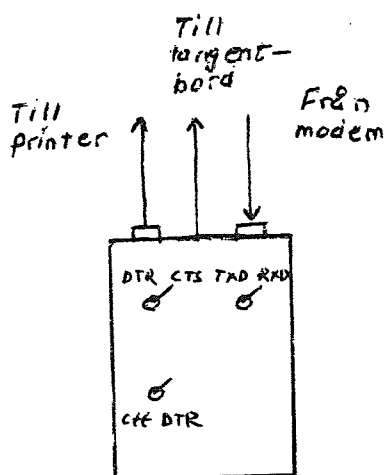
Det är även ganska lätt att modifiera testprogrammet så att det på skärmen skriver ut de radnummer som testas och innehållet i den byte som testas av programmet. Testningen av 32 k RAM tar nämligen en försvarlig tid. Inte går det fortare om minnesinnehållet skall skrivas på skärmen, men man kan följa med och se hur programmet arbetar och är säkrare på att "något testas".

Tack
gläd
och
Jag
janu
kom
förMå
från
stor
publDet
men
skal
vadVad
Unde
janu
troihar
tro:
kuniDet
ber:
mån
tån
avDet
kan
En
nomffe
ssa
ast
stoen
Vid
cus
arc
att
Det
arc
sseNi
någ
mieJöi
Bi
175
075Mec
As-
Sci
Sve

TERMINALKOPPLING

Du som vill utnyttja Gunnar Tidners program för att hämta program eller datafiler från en annan ABC 80 eller från stordator t.ex. DEC-10 hos QZ i Stockholm, kan ha behov av följande kopplingsanvisning för att kunna köra printern parallellt med bildskärmen.

En parallellkoppling är nödvändig för printrar som kommunicerar via V24-utgången, eftersom modemmet är anslutet till denna utgång.



Inkopplingen kan lätt göras i en liten plast låda ca. 9x12 cm i vilken fästs två niopoliga modemkontakter (chassiekontakter) för lödanslutning. Lödningen kan sedan ske direkt på kontaktstiften. Ovanpå lådan monteras de tre brytarkontakter som erfordras

Printersladden måste lödas ihop liksom modemsadden. Ut till datorns tangentbord dras en sladd, 7 polig, vilken i lådan kan lödas direkt till lödstiften på modemkontakterna.

Printersladden erfordrar en 9 polig och en 25 polig modemkontakt. Den vanliga printersladden kan inte användas då den kortsluter modemkopplingen. Kopplas den om går den inte att använda för vanliga utprintningar. Man måste alltså hålla sig med 2 olika sladdar för att kunna köra printern även i terminalkoppling.

Modemsadden behöver ej kopplas om vare sig man sätter den i tangentbordet (V24 kontakten) eller i kopplingslådan. Den erfordrar en 9 polig modemkontakt samt en 25 polig modemkontakt hane (stift).

En 2 polig brytare on-off-on används för att växla mellan TD (transmit) och RXD (receive) beroende på om printern skall skriva ut vad någon annan gör för bearbetningar i din dator (TXD) eller det du själv får upp på din bildskärm och vill dumpa ut på printern (RXD).

En likadan brytare används för att släppa fram DTR-signal (Data Terminal Ready) till din printer eller för att sända CTS-signal (Clear To Send).

Genom att använda ovanstående koppling får man även möjlighet till automatisk nedkoppling av telefonlinjen - om modemmet står i Auto-läge - efter det att den uppringande har loggat ut.

En enpolig brytare on-off används för att skicka spänning till modemmet respektive bryta ned telefonlinjen när man loggat ut från en annan dator.

Följande komponenter erfordras.

- 3 niopoliga modemkontakter (honor) med plastkåpor
- 1 25 polig modemkontakt (hona) med kåpa
- 1 25 polig modemkontakt (hane) med kåpa
- 4 m sladd 7 polig
- 2 niopoliga modemkontakter (hanar för chassiemontage)
- 2 tvåpoliga brytarkontakter on-off-on
- 1 enpolig brytarkontakt on-off
- 1 gumnigenomföring för sladd
- 4 M3 skruvar med muttrar och brickor kopplingsladd
- 1 låda ca 9x12 cm

Plastlåda bör väljas då hålen för modemkontakterna kan vara besvärliga att ta upp.

Date
Stift

1 0
2 0
3 0
4 0
5 0
6 0
7 0
8 0

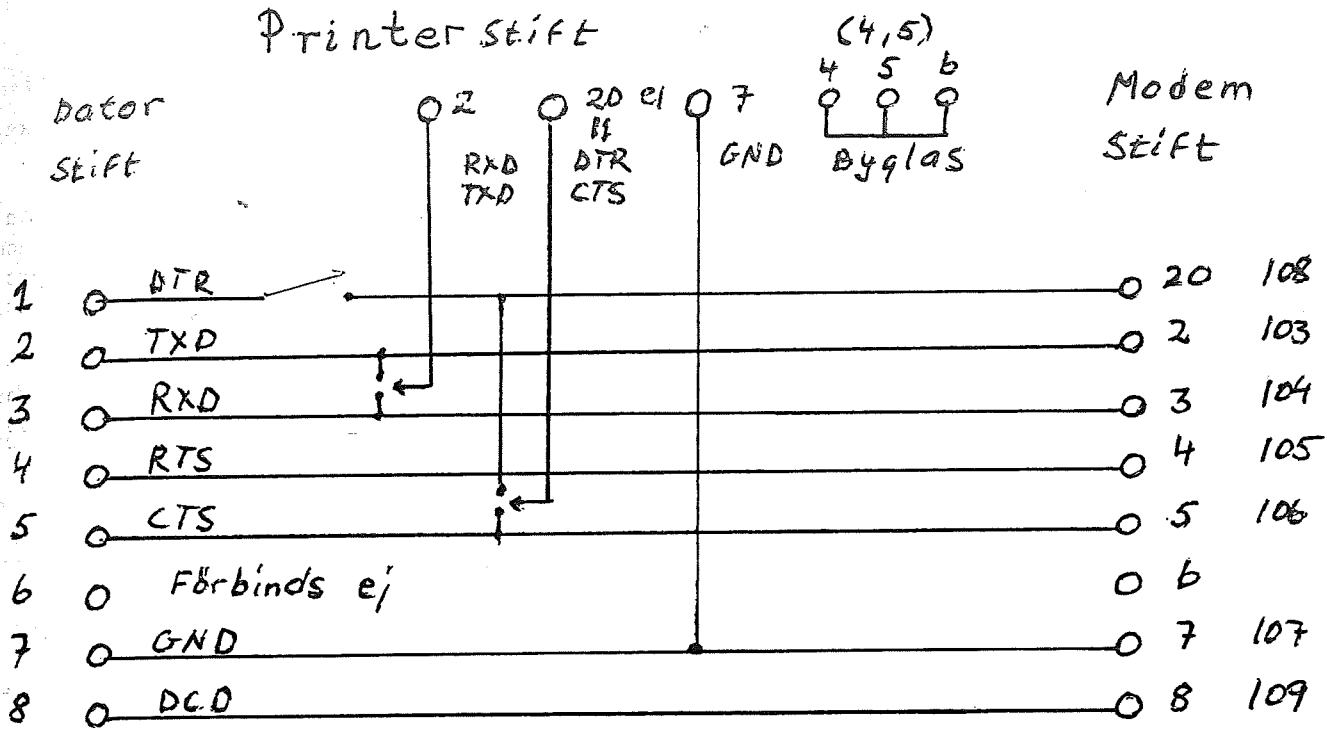
Kc
Vi
ma

Kolla i Di
kommando I
ställa vär
escape. C
ginalen h
Du har an

PROGRAMLI

1 OPE
2 ; #
3 ; "
UT
4 ; #
7,5
5 CLC
6 ENT

Printerstift



Kommentar 1982:

Vilka stift som skall byglas är beroende på vilken skrivare man avser. Exemplet avser en Diablo 1650.

NAGOT

Kolla i Din printermanual vilket escape-kommando Din printer behöver för att ställa vänstermarginalen. CHR\$(27)=escape. CHR\$(27,57) ställer vänstermarginalen hos Diablon i den kolumn, K, som i har angett i programmet.

PROGRAMLISTA VMARG.BAS

```

1 OPEN "PR:" ASFILE 1
  ; #1,CHR$(27,69)
3  ; "VÄNSTERMARGINAL I KOL "; : INP
  UT K
4  ; #1,CHR$(27,9,K+1); : ; #1,CHR$(27,57)
5  CLOSE 1
6  END

```

Nästa program, av samma kategori, visar hur ett maskinspråksprogram kan skapas. Genom att med instruktionen POKE(peta dit) lägga in ett antal decimala värden från adress 65408, alltså i POKE-arean (se bruksanvisning till ABC 80), och framåt.

Varje decimalvärde representerar en maskinkodsinstruktion, som enkelt körs med hjälp av CALL-instruktionen till den aktuella adressen. OBS! POKE -2 är samma som POKE 65536-2. Vad programmet gör? Jo, det listar helt enkelt sig själv på skärmen.

```

10 POKE 65408%,42%,254%,255%,126%,61%,200%,17%,64%,254%,213%,205%,149%,17%,34%,254%,255%
20 POKE 65424%,235%,209%,167%,237%,82%,68%,77%,235%,205%,168%,1%,195%,56%,7%
30 POKE -2%,PEEK(65052%),PEEK(65053%)
40 IF PEEK(CALL(65408%))-1% THEN 40

```

Roliga rutan

Här presenteras program av mer underhållande karaktär. Det kan alltså röra sig om både spel och program som inte utför någon direkt användbar funktion, men kanske är intressant ur programteknisk synpunkt.

Här kommer just ett program som är intressant programtekniskt. ABC 80's basic är både kraftfull och snabb, även om den fortfarande kan förbättras. Detta exempel visar hur ett program med många funktioner, kan skapas i en enda program rad.

```
1 IF A%-1 IF A% A%=A%-1 : Z=Z*A% : GOTO 1 ELSE INPUT A% : Z=A% : GOSUB 1 : ; Z : A%=0 : GOTO 1 ELSE A%=0 : RETURN
```

Det kan kanske vara på sin plats med en förklaring av en del av programfunktionerna. Instruktionen IF A% ger till exempel träff, d.v.s. programraden exekveras, om A% innehåller ett värde skilt från noll. IF är föresten en instruktion som ställer problem för en del. Det är svårt att följa logiken och veta när en IF-sats utförs. Generellt gäller ju att, är villkoret sant (träff) vid en IF-test, utförs de instruktioner som följer, även om flera följer på varandra, med kolon emellan. Är villkoret ej sant, sker hopp till nästa instruktion. Läger vi till ELSE, efter sista instruktionen, följt av ytterligare instruktioner, utförs dessa istället för hoppet till nästa instruktion. I exemplet ovan, har vi 2 IF och ELSE satser, då blir det värre. Vi gör om programmet lite, och tittar på det.

```
IF(1)A%-1 IF(2)A%..ELSE(2)..ELSE(1)..
```

Då ser vi att om IF(1) är falsk, så fortsätter vi med instruktionen efter ELSE(1). Om däremot IF(1) är sann, provar vi om IF(2) är sann. Är IF(2) falsk, fortsätter vi med ELSE(2), annars utförs instruktionerna efter IF(2). I båda fallen, måste vi avsluta med en GOTO instruktion, för att ej komma över på (1)'s domäner, eller hopp till nästa instruktion.

Kunde denna lilla förklaring kasta ett förklarande sken över vad programmet gör.

```
* RITOSUDD *

Programmet är självinstruerande och kan laddas in med användning av en maskin. Bilderna kan ej sparas på tape eller flexskiva men detta programavsnitt kan skrivas till av den som så önskar genom att lägga till de delar av programmet som sparar radinnehåll respektive läser in radinnehåll till minnet.

Observera att de rader som skall sparas respektive läsas tillbaka är bildminnet. Detta finns på adress 31744 till 32767.

Följande kod är skriven i BASIC (BA) och kan laddas in med hjälp av en maskin. Ange från minnesadress ENTER (EN): Vata in maskin EXECUTE (EX) Kör maskin tangentsbord SAVE (SA): Spara maskin flexskiva. LOAD (LO): Ladda ett program till minnet. BASIC (BA): Återgå till BASIC handboken för form varför till och från ledning av program som skrivna i BASIC adress 654C kan ändras anvisningen till hexpok programmet göra utskrift och som kör poke-arean

10 REM *** LASSES RITBORD MED RADER ***
20 ; CHR$(12)
30 ; 'LASSES RITBORD MED RADER FUNKTIONER'
40 ; '=====SAVE (SA):=====
50 ; 'FÖRFLYTTNING UPP=U : NED=N'
70 ; 'VÄNSTER=H HÖGER=J'
80 ; 'RITA=Å+R SUDDA=Å+S'
90 ; 'ÄVEN DIAGONALA RÖRELSER=Y/I/B/SKIVA.
100 ; 'Ä BETYDER ÄNDRA MOD.'
110 ; 'TRYCK NÅGON TANGENT FÖR START!'
120 ; : GET A$
130 ; CHR$(12) : FOR I=0 TO 23
140 ; CUR(I,0);CHR$(151);
150 NEXT I
160 GET Q$ : REM S=SLÄCKA .... R=RITA
170 GET Å$
180 IF Å$='N' L=L+1
190 IF Å$='U' L=L-1
200 IF Å$='J' K=K+1
210 IF Å$='H' K=K-1
220 IF Å$='I' L=L-1
230 IF Å$='I' K=K+1
240 IF Å$='B' K=K-1
250 IF Å$='B' L=L+1
260 IF Å$='Y' K=K-1
270 IF Å$='Y' L=L-1
280 IF Å$='M' L=L+1
290 IF Å$='M' K=K+1
300 IF Å$='Å' GOTO 160 : REM ÄNDRA MOD. R/S
310 X=35+L
320 Y=40+K
330 IF X>69 GOSUB 400
340 IF X<1 GOSUB 400
350 IF Y>77 GOSUB 400
360 IF Y<3 GOSUB 400
370 IF Q$='R' SETDOT X,Y
380 IF Q$='S' CLRDOT X,Y
390 GOTO 170
400 L=0 : K=0 : RETURN
```

* MONITOR *

Monitor är ett program med vilket du kan
ladda in maskinspråk (hexkod) i ABC-80,
lägga maskinspråksprogrammet på flexski-
va eller kassett, läsa in programmet,
testköra det, dumpa ut programmet i hex-
adecimal kod eller vilken del som helst
av minnet på skrivare.

kti

Följande kommandon finns, varav endast
de två första tecknen behöver anges:

net

67. PEEK (DU):

Läs från - till vilken hexadecimal
minnesadress utskrift skall ske.

ENTER (EN):

Mata in maskinkod i minnet.

EXECUTE (EX):

maskinkodsprogram som matas in från
tangentbord, tape eller flexskiva.

---SAVE (SA):

Spara maskinkodsprogrammet på tape eller
flexskiva.

LOAD (LO):

Ladda ett program från tape eller flex-
skiva.

BASIC (BA):

Återgå till basic.

RTI

ITA Basic handhar numeriska värden i decimal
form varför programmet gör omvandling
till och från hexkod. Detta görs med an-
ledning av att de flesta maskinspråks-
program som finns i tidskrifter m m är
givna i hexadecimal form. Lagring av
maskinkod sker i poke-arean (decimal
adress 65408 - 65535). Lagringsadressen
kan ändras i programmet (se även bruks-
anvisningen till ABC-80, i anslutning
till hexpoke-programmet). Ändring av
programmet är nödvändig för den som vill
göra utskrift på printer av maskinkod
och som kör en printerrutin som använder
poke-arean som utskriftsbuffer.

MOI

Ändringen görs på programrad 400, 470
och 780 i programmet. Utskrift på skär-
men kan erhållas genom att ändra alla
41, mot t.ex. #P%, och lägga till några
programrader i början av programmet som
frågar efter utskrift t.ex

1 P%=0%

2 ;"Skall utskrift ske på printer
J/N) ?";

3 GET F\$:IF F\$="J" OR F\$="j"THEN 14
ELSE 20

4 P%=1%

20 OPEN "PR:" ASFILE P%

Vill Du länka ett maskinkodsprogram till
ett basic-program kan maskinkoden hämtas
genom CALL-kommandot i basic exempelvis
CALL(65408%). Maskinkoden måste avslutas
med hexadecimalt C9 (RET) för att åter-
hopp skall ske till basic.

Sedan Du knappat in programmet och lag-
rat det på tape eller flexskiva skriver
Du RUN och får då upp meny på skärmen.

Slå in EN för att mata in ett maskin-
kodsprogram. Ett frågetecken kommer upp
på skärmen varefter Du kan mata in den
hexadecimala koden, två siffror i taget
följt av RETURN. Slå in bokstaven O samt
tryck RETURN för att avsluta programmet.

Dump-kommandot skriver ut minnet hexa-
decimalt från angiven startadress till
den slutadress som anges. Observera att
programmet avslutas med att skriva ut
alla 16 bytes som finns på den sista ra-
den i minnet.

Pröva att slå in följande lilla program
som visar var stacken slutar efter åter-
hopp till basic.

21 00 00 LD HL 0000H

39 LD HL,SP

C9 PET

Lagra programmet från adress FF80 till
FF84. Skriv därefter NEW för att ta bort
monitor-programmet och skriv in följande
basicprogram.

10 ;"Stackens slutadress":;

20 A%=CALL(65408%)

30 ;65536+A%

40 END

Spara programmet på tape eller skiva.
Stacken börjar på adress 62720 decimalt
och växer nedåt. Med dessa två program
kan man kontrollera hur stor stacken är.

Kör monitorprogrammet och ladda in ma-
skinspråksrutinen i minnet, skriv NEW
och ladda därefter in det lilla basic-
programmet på vanligt sätt och skriv
RUN. På skärmen får Du då upp adressen
där stacken slutar.

Ett så här kort maskinkodsprogram kan
lika gärna laddas i decimal form genom:
POKE 65408,33,0,0,57,20!

red

PROGRAMLISTA MONITOR.BAS

10 REM COPYRIGHT ODD ROLANDER,1978-12
-20

20 OPEN "PR:" ASFILE 1

A B C-bladet Samlingsnummer
1980 : 2 sidan 20

<pre> 30 ; CHR\$(12) 40 DIM A\$(16),S(16) : A\$="0123456789A BCDEF" 50 ; TAB(11)"*** MONITOR ***" : ; 60 ; : ; "VÄLJ: DUMP,ENTER,EXECUTE,SA VE,LOAD ELLERBASIC" 70 INPUT C\$ 80 IF LEFT\$(C\$,2)="DU" THEN 150 90 IF LEFT\$(C\$,2)="EN" THEN 390 100 IF LEFT\$(C\$,2)="EX" THEN 460 110 IF LEFT\$(C\$,2)="SA" THEN 500 120 IF LEFT\$(C\$,2)="LO" THEN 730 130 IF LEFT\$(C\$,2)="BA" THEN END 140 GOTO 60 150 ; #1,TAB(25)"MONITOR" : ; #1,"" 160 ; : ; "ANGE START OCH SLUTADRESS I HEX." 170 ; "STARTADRESS FÖR DUMP " : INPUT H\$(1) 180 ; "SLUTADRESS FÖR DUMP " : INPUT H \$(2) 190 ; #1,"STARTADRESS,HEX ";H\$(1) : ; #1,"" 200 ; #1,TAB(11)"1";" 2";" 3";" 4"; " 5";" 6";" 7";" 8";" 9";" A ";" B";" C";" D";" E";" F" : ; #1,"" 210 FOR I=1 TO 2 220 H\$=H\$(I) 230 GOSUB 910 240 D(I)=D 250 NEXT I 260 FOR I=D(1) TO D(2) STEP 16 270 D=I 280 GOSUB 1010 290 ; #1,MID\$(A\$,H4+1,1);MID\$(A\$,H3+1, 1); 300 ; #1,MID\$(A\$,H2+1,1);MID\$(A\$,H1+1, 1);" "; 310 FOR J=0 TO 15 320 D=PEEK(I+J) 330 GOSUB 1010 340 ; #1,MID\$(A\$,H2+1,1);MID\$(A\$,H1+1, 1);" "; 350 NEXT J 360 ; #1,"" 370 NEXT I 380 GOTO 60 390 ; "LAGRING SKER I POKE-AREAN 65408 -65535" 400 FOR I%=65408% TO 65535% 410 INPUT H\$ 420 D%=INSTR(1%,A\$,LEFT\$(H\$,1%)) 430 IF D%=0% THEN ; "SLUTADRESS: ";I%+ 1% : GOTO 60 440 POKE I%,(D%-1%)*16%+INSTR(1%,A\$,RI GHT\$(H\$,2%))-1% 450 NEXT I% 460 ; 470 A%=CALL(65408) 480 ; A% 490 GOTO 60 </pre>	<pre> 500 ; "STARTADRESS FÖR SAVE I HEX " : INPUT H\$ 510 GOSUB 910 520 D1=D 530 ; "SLUTADRESS FÖR SAVE I HEX " : INPUT H\$ 540 GOSUB 910 550 D2=D 560 ; "PROGRAMNAMN " : INPUT V\$ 570 PREPARE V\$ ASFILE 2 580 FOR I=D1 TO D2 STEP 16 590 FOR J=0 TO 15 600 S(J+1)=PEEK(I+J) 610 NEXT J 620 FOR Ö=1 TO 8 630 ; #2,S(Ö) 640 NEXT Ö 650 FOR Ö=9 TO 16 660 ; #2,S(Ö) 670 NEXT Ö 680 NEXT I 690 ; #2,999,0,0,0,0,0,0,0 700 ; #2,0,0,0,0,0,0,0,0 710 CLOSE 2 720 GOTO 60 730 ; "STARTADRESS FÖR LOAD I HEX " : INPUT H\$ 740 GOSUB 910 750 ; "PROGRAMNAMN " : INPUT V\$ 760 OPEN V\$ ASFILE 2 770 ONERRORGOTO 890 780 FOR I=D TO 65535 STEP 16 790 FOR Ö=1 TO 8 800 INPUT #2,S(Ö) 810 NEXT Ö 820 FOR Ö=9 TO 16 830 INPUT #2,S(Ö) 840 NEXT Ö 850 FOR J=0 TO 15 860 POKE I+J,S(J+1) 870 NEXT J 880 NEXT I 890 CLOSE 2 900 GOTO 60 910 D=0 : RESTORE 920 Z=LEN(H\$) 930 FOR K=Z TO 1 STEP -1 940 READ M 950 FOR J=1 TO 16 960 IF MID\$(H\$,K,1)=MID\$(A\$,J,1) THEN X=J-1 : J=16 970 NEXT J 980 D=D+X*M 990 NEXT K 1000 RETURN 1010 H4=INT(D/4096) 1020 H3=INT((D-H4*4096)/256) 1030 H2=INT((D-((H4*4096)+(H3*256)))/16) 1040 H1=D-((H4*4096)+(H3*256)+(H2*16)) 1050 RETURN 1060 DATA 1,16,256,4096 </pre>	<pre> Programme kortning: PROGRAMME 10 ; C 20 ; C 30 ; C 40 ; C 50 FOF 60 ; C 70 DIM 80 RAM 90 FOF 100 REA 110 NEX 120 ; I IG 130 ; I INC 140 ; I === 150 ; : 180 N=C 190 N=N 200 I=I 210 I=I 220 Q=C 230 GOS RKO: : 240 INP 250 GOS 260 ; 270 IF : 280 ; : 290 Q=Q 300 GOT 310 R=8 320 R=I 330 IF 1 340 IF 1 350 IF 1 360 IF 1 370 ; 1 380 ; 1 0 390 ; 1 190 400 ; 1 0TO 410 ; 1 420 ; 1 1 430 ; 1 1 </pre>
---	--	---

24 K BASIC

Några programmeringstips

Om du tillhör de som tycker om att programmera i maskinspråk så stöter du förr eller senare på problem att få ut värden från maskinspråksprogrammet till t ex en printer eller bara på skärmen. Även att få in data från tangentbord kan ställa till problem. BASIC-tolken innehåller ju rutiner för detta och mer, så varför inte använda dessa, skrev Gunnar Hellquist i Skene till redaktionen. Här kommer därför en Gunnars lista på olika användbara rutiner. Rutinernas anropsadress står i decimal form.

1 UTMATNING

Rutinen för utskrift på skärmen börjar på adr 424. Indata ges i reg BC som talar om många tecken som ska skrivas ut och HL som pekar på första tecknet i strängen. Olika subrutiner som används är töm skärm (adr 630), scrolla (adr 564), beräkna bildadress (659). Utskrift på skärmen börjar på den rad och kolumn som specificeras av pekare på adress 65011,65012. Det finns en möjlighet att skriva ut på en fil, då man bekant skärmen har filnummer 0. Den rutinen har startadress 2260 och anropar lämpliga rutiner för utskrift på det filnummer som specificeras av pekare på adress 65058. Här måste dock filen vara öppnad från BASIC (OPEN "PR;" ASFILE 1) innan anrop sker, annars fås felutskrift. Även här ges data i BC (längd) och HL (start).

2 INMATNING

Inmatning från tangentbord sker med den rutin som används vid ED på BASIC-rader. Den här rutinen har två inhoppadresser beroende på önskad funktion. Vid inhopp på adress 682 fås en funktion liknande INPUT-LINE i BASIC. Indata i HL (startadress för sträng som tillverkas) och reg C (maxlängd på sträng). Även här sker utskrift på skärm enligt pekare 65011, 65012. Vid inhopp på adress 685 ges möjlighet att editera redan tillverkad sträng i enlighet med ED. Nödvändig data utöver tidigare nämnda är DE (adress insträng). Insträngen ska sluta i tecknet 13 (vagnretur). Räkna med att alla register, utom IY, i Z80 påverkas av ovanstående operationer. Även vid inmatning finns en rutin för inmatning från valbar fil. Den har startadress 2269. Filnummer i adress 65058. Se i övrigt ovanstående.

3 OMVANDLING

I BASIC-tolken finns också rutiner för omvandling från flyt/heltal till text och vice versa. Den mest använda rutinen är antagligen från heltal till text. Den börjar på adress 6229 och kräver data HL (binärt tal) och DE (startadress för utskrift). Alternativt kan inhopp ske vid 6242, vilket ger utskrift utan minustecken mellan 0 och 65536. Vid inmatning av tal kan rutinen på 6209 (med tecken) alternativt 6174 (utan tecken) användas. Indata HL (start text som skall omvandlas till heltal), utdata DE (bin heltal). CARRY flagga sätts vid fel på insträng.

Eftersom ni i ett tidigare nummer frågade efter önskemål på en eventuell ny BASIC så skriver jag här en del helt egna önskemål.

- *Möjlighet att skicka över fler argument vid CALL. Helst skulle jag vilja kunna ladda upp samtliga register i CPU-n från ex ett antal minnesplatser. ONERRORGOTO bör fungera på fler feltyper text:ERR 5,9,10,32. Åtminstone bör finnas en möjlighet att kontrollera om en fysisk enhet, tex PR: finns ansluten och försedd med drivprogram, utan att trilla ur programmet.
- *Möjlighet till restart eller continue.
- *Någon slags Format- eller Print using för formattering av utskrift.
- *Gärna en liten 'monitor' för laddning, verifiering och provkörning av maskinspråksprogram.

LUXOR:s återförsäljare

Med tanke på den dåliga kontroll LUXOR har över sina återförsäljare på ABC80 är det enligt min mening dags att starta en debatt. Idag är tyvärr situationen att vilken radioaffär eller vilket varuhus som helst kan ta hem en ABC80 och börja sälja. Tyvärr får kunden ingen hjälp med utval av maskin och mjukvara. Ofta är situationen till och med den att personalen på sin höjd kan demonstrera DEMO-programmet. Även om manualerna är välskrivna och relativt lättlästa så uppstår problem som kunden ej tänkt på. T ex vissa exemplar av FD 2 har en otrolig vana att krångla på mystiska sätt. Tydligt har en del av dessa minskat på senare tid, men situationen kvarstår fortfarande att många kunder måste ha blivit avskräckta från inköp pga dåliga erfarenheter.

Om ett mindre företag planerar att lägga över bokföringen på ABC80 uppstår ofta problem i samband med de manuella rutinerna. Orsaken är som regel att ingen

person meddelat hur viktigt det är att se över hela rutinerna i samband med en sådan här övergång.

Även om det lyckligtvis inte är ofta så händer det att det är fel i programvara. En okunnig kund kan lätt missledas att tro att felet ligger i maskinvaran om han ej kan få hjälp. För att garantera köpare/användare av ABC80 en god service och hjälp vid fel måste vi från klubbens sida ställa krav på LUXOR om en ordentlig politik för återförsäljare. Vi måste av en återförsäljare kräva åtminstone minimum-kunskaper i feldiagnostik (hård-mjukvarufel) och grundläggande kunskaper om inbyggda begränsningar. Framför allt vid försäljning till personer utan större datorkunskaper måste återförsäljaren erbjuda en möjlighet till kontinuerlig uppföljning av system-funktion, eventuella problem och information om tillgänglig mjukvara.

MISSTÄNKT FEL I ASSEMBLER (KASSETTEVER.)

Ex: assemblering av LD BC, (FF00) ger gallmatias. Felet ligger i filen på band som innehåller mnemonics. Följande ändring medför en förbättring: 12:e LD LD, 60744, 50220 bytes mot LD, 60747, 50220

GUNNAR HÄLLQVIST

SAMMANSTÄLLNING

KOMMANDON	*GOTO, GOSUB	Ex. GOTO 220, startar programmet vid rad 220
	*CONT	Fortsätter programmet utan att nollställa variablerna
	*DELETE	Ex. DELETE 40-50, tar bort alla rader inom det definierade intervallet
	*STEP	Vid STEP-mod exekveras en BASIC-rad i gången
	*FOR - NEXT	Vanliga BASIC-satser även som kommandon
CTRL-TANGENTERNA	*CTRL-^	Töm skärmen (; CHR\$(12))
	*CTRL-'	Starta bandspelaren (RUN CAS:)
	CTRL-	Starta bandspelaren (MERGE CAS:)
VARIABLER	*TIMES	Tiden i hhmss-format
	*BYTE	Antal lediga bytes
FUNKTIONER	ASN(X)	arc sin x
	ACS(X)	arc cos x
	DEG(X)	180*X/PI=57.29577951*X
	RAD(X)	PI/180*X=0.01745329252*X
	*MOD(A,B)	A-INT(A/B)*B (resten; MOD syftar på 'modulo')
	*MAX(X,Y)	-(X>Y)*X-(Y>X)*Y
	*MIN(X,Y)	-(X<Y)*X-(Y<X)*Y
	SUM(X)	Summan av elementen i vektorn (matrisen) X
	PRD(X)	Produkten av elementen i vektorn (matrisen) X
	LOG2(X)	LOG(X)/LOG(2)=1.442695041*LOG(X)
ONERRORGOTO	*ERR 9	Index utanför strängen
	ERR 10	Texten får ej plats i strängen
	ERR 50	Kvadratrot ur negativt tal
	*ERR 62	DOT-adress utanför skärmen
	ERR 65	Spill i ASCII-aritmetik
	*ERR 66	Sträng ej numerisk
ÖVRIGT	*10 siffrors noggrannhet	
	Möjlighet att lagra variabler i flerdimensionella matriser	

* Stjärnförsedda rader anser undertecknad vara speciellt viktiga!

Så här långt våra läsare. Vi på redaktionen har låtit såväl Lars Karlsson, Dataindustrier AB, och Martin Graap vid Luxor Division Datorer uttala sig om just utbyggnad av ABC-Basic.

Frågan om ett 24k Basic-ROM har diskuterats, men för närvarande är det inte aktuellt med någon förändring av befintlig 16k ROM. Det mesta pekar mot att en kraftfullare Basic först lanseras i samband med marknadsintroduktionen av den stundande nya generationen datorer. En sådan maskin kommer troligen att vara utrustad med såväl 24k Basic som färg- och högupplösningsgrafik. Något annat vore ur marknadsföringssynpunkt komplett vansinne, berättar Martin Graap för ABC-bladet.

Samma inställning har Lars Karlsson vid Dataindustrier AB, och tillägger: En större och kraftfullare Basic finns redan framtagen, men vi avser inte att släppa den till ABC 80. Däremot är det aktuellt att använda en större Basic i kommande generationer av datorer.

SKIVMINNEN

Vid vårt samtal med Martin Graap togs också upp frågan om vad som komma skall på skivminnessidan. Här lät Martin meddela att "innan jul kommer Luxor med egentillverkad enhet kapabel till dubbel packningstäthet. Varje drivenhet kommer dessutom att vara omkopplingsbar mellan enkel och dubbel packningstäthet för att äldre programskivor fortfarande ska kunna användas. Något beslut om att hjälpa kunder med konvertering av äldre program till dubbel packningstäthet har dock inte gjorts."

Som läget är idag kommer vi således att erbjudas ett flertal olika typer av skivminnen, bl a från Scandia Metric, Sattco och Luxor. Samtliga leverantörer har aviserat de nya skivminnestyperna till senhösten.

Frågan om ett 24k Basic-ROM har diskuterats, men för närvarande är det inte aktuellt med någon förändring av befintligt 16k ROM, säger marknadsföringschef Martin Graap vid Luxor Division Datorer.

TEAM 100

Så heter Luxors nya satsning på marknadsföringen av ABC 80. Tanken är att de bästa, mest aktiva återförsäljarna ska ingå i en grupp - Team 100. Denna grupp ska aktivt stödjas av tillverkaren.

Aktiviteter har redan kommit igång. I början av september träffades "de hundra" på en konferens i Finland, där Luxor också passade på att presentera höstens kommande nyheter (se marknadsidorna).

Det intressanta för konsumenten är att alla återförsäljare kommer att få en direkt stöd från Luxor, vilket innebär bättre service till kunderna. Omfattande utbildningsverksamhet för både säljare och tekniker är redan igång. Dator teknik, systemutbyggnad och expansion och programmeringsprinciper går genom. Speciella demonstrationspaket kommer att tas fram för att på ett bättre sätt presentera större, mera invecklade program.

Säljarna uppmärksammas även på de ergonomiska problemen och lämpliga hjälpmedel kommer successivt att introduceras i deras sortiment. En förteckning över och beskrivning utav alla nya program ska regelbundet sändas till alla återförsäljare. Kunden garanteras att vid köpet få den senast uppdaterade versionen av programmet.

Vi hoppas att snart få se Team 100-märket ute på sta'n, för enligt Luxor betyder det bättre sortiment, bättre service och snabbare leveranser.

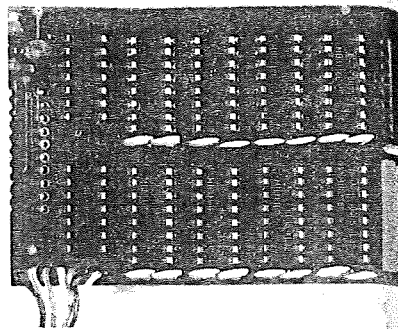
KURVPLOTTNING PÅ HOUSTON-PLOTTER

Tidningen har testat ett program från Databutiken i Uppsala som kan användas för att plotta kurvor på en Houston-plotter. Programmet, som heter Kurvette, kräver flexskiveenhet och 32k minne för att kunna användas, printer (om värden skall skrivas) samt plotter (om diagram skall plottas). Programmet beräknar olika regressionslinjer och kan rita ut diagram både på ABC'ns bildskärm och på plotter.

Kurvette beräknar regressionskoefficienter, korrelationskoefficienten och determinationskoefficienten för anpassning av datapar enligt minsta kvadratmetoden. Dessutom plottar programmet inmatade värden samt regressionslinjen. Programmet kan beräkna linjär, hyperbolisk, exponentiell, logaritmisk, geometrisk och parabolisk regression. Multipel linjär regressionsanalys klarar dock programmet inte. Som helhet fungerar Kurvette väl och klarar att behandla maximalt 100 inmatade datapar. Inmatning kan ske antingen direkt från tangentbordet eller från en fil. Allt för stora tal, d v s tal som ABC-80 uttrycker med exponent, kan inte matas in. Utskrift kan ske i diagramform och med text på axlarna, samt av funktionen och korrelationskoefficienten. Programmet beräknar självt optimala värden för skalning av diagrammet på bildskärmen och på plottern. Man kan emellertid själv ändra skalningen på plottern om man så önskar.

För att plottning skall kunna ske måste programmet "PLOT" ha laddats (drivprogramvara). Av tidsskäl har inte analys gjorts av hur stor noggrannhet i beräkningarna programmet klarar, men ett exempel på linjär regression vid fastighetsvärdering har körts med korrekt resultat.

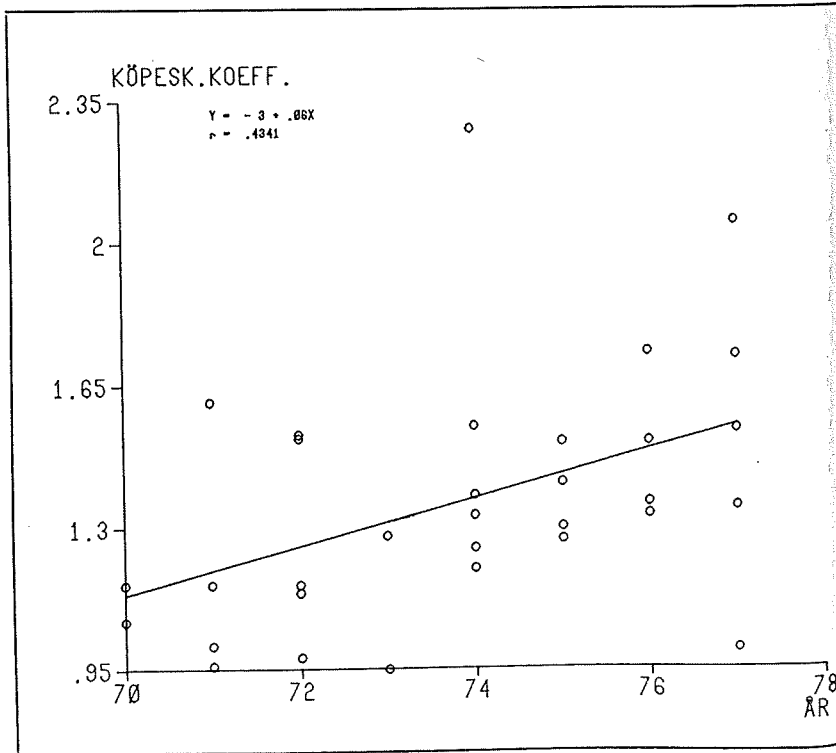
Programmet är värdefullt även för den som inte har plotter utan hänvisas att köra enbart på bildskärm (och printer). Noteras bör att bildskärmens grafik kan dumpas ut i tex på den lilla printern Microline 80. Databutiken i Uppsala bör därför ta fram en specialversion av programmet för detta ändamål. Plottningsrutinen är nämligen skyddad och kan inte ändras av användaren.



BÄTTRE MINNE

Oräknerliga är de tillfällen då ABC-80 arbetsminne tycks för litet. Ett flertal olika expansionsminnen finns tillgå på marknaden. Vi på ABC-bladet har provat RAMEX 80 på 16k byte från Databutiken i Uppsala. Hela inbyggnadskortet, som skall monteras i tangentbordet, är placerat på ett väldigt nerat och snyggt dubbelsidigt glasfiberkort. Själva monteringen är väl genomtänkt och görs genom att placera expansionskortets anslutningsstift direkt i en av ABC-80's dil-socklar. Förutom denna anslutning ska sju olika trådar lödas samt en speciell kontakt fästas i ytterligare en dil-sockel. Totalt skall 2 kretsar på ABC-80's grundkort lossas och flyttas över till expansionskortet.

Hela monteringsmanövern tog oss knappt en halvtimme och var som helhet mycket enkel att utföra. För att undvika risk för felvändning av de två IC-kretsar som placeras på expansionskortet borde dock ritad skiss bifogas med RAMEX 80. I övrigt finns inga invändningar mot de monteringsanvisningar som Databutiken skickar med det extra minneskortet. RAMEX 80 kostar för närvarande 1795 kr inkl. moms (20,63% i 1980 kr inkl. montering, då säljaren övertar garantin för tangentbordet, och kan rekvireras direkt från Databutiken i Uppsala, tfn: 011 70 60 /61 /62.



TEST AV MJUKVARA

METRIC MINI PASCAL

Pascal är ett kraftfull programmeringsspråk som av bl a leverantörerna börjar lanseras till ABC-80. Vi har på tidningen testat Scandia Metric's Mini Pascal som inte är en fullständig Pascal, utan som namnet anger en minivariant av Pascal. En större Pascal är under framtagande och kommer att kunna köras även den på ABC-80. För att kunna köra Mini-Pascalen fordras att man har tillgång till flexskiveenhet. Pascalen laddas in i minnet från flexskivan vilket innebär att pascalinterpretatorn tar viss del av ramminnet i anspråk. Det går alltså inte att kasta ut Basicen som ligger i Rom-kapslarna i botten. Tillgängligt programmeringsutrymme minskar dels härigenom och dels genom att vid kompileringen måste pascal-kompilatorn få plats i minnet för att få en exekverbar kod. I gengäld är Pascalen kraftfullare än Basicen.

VARFÖR PASCAL ?

Pascal har trevliga egenskaper. Block-struktur, namngivna procedurer och funktioner, lokala variabler och avancerade datastrukturer. Programmen är lättlästa och självdokumenterande. Pascal brukar därför användas för att beskriva algoritmer. Basic saknar nästan alla dessa egenskaper. Pascal anses speciellt lämpligt för undervisning genom sin strikta uppbyggnad som uppmuntrar till goda programmeringsvanor. Metric Mini Pascal är tänkt att användas i undervisningssammanhang men anses olämplig för produktionsändamål. Kompilatorn är överifierad och saknar filhantering. Endast heltal och heltalsvektorer är medtagna. Datatypen Real för flyttal finns alltså inte alls. I Minipascalen saknas GOTO. Alla övriga styrande satser är likadana som i äkta Pascal med något undantag.

SPECIELLT I MINIPASCALEN

För samtliga aritmetiska operationer gäller att "overflow" inte fångas upp. Talområdet är begränsat till -32 768, + 32 767. Matriser är begränsade till endimensionella vektorer av heltal. Index börjar alltid på 0. Parametrar i minipascalen kan inte vara VAR-deklarerade och den kontrollerar inte om en funktion eller procedur är deklarerad tidigare i samma block. Man kan alltså råka ha två funktioner eller procedurer med samma namn och anropa den ena fast man egentligen ville anropa den andra. Den kontrollerar inte att antalet parametrar i deklARATIONEN och anropet av en funktion eller procedur stämmer överens. Detta kan leda till svårupptäckta fel.

In och utmatning sker i en något förnklaad form i minipascalen än i vanlig Pascal.

READ och WRITE-satserna kan läsa in respektive skriva ut värden av variablerna i heltalsform, hexadecimalform eller som ASCII-värden. WRITE får innehålla funktionsanrop och teckensträngar.

Storleken av matriser kontrolleras inte någonstans. Alla reserverade ord i Pascal finns inte och dessa är unika bara till 5:e bokstaven. Man kan således inte definiera funktionen WRITELN för kommunikation med en skrivare eller terminal. Identifierare i övrigt är signifikanta t o m 8:e bokstaven.

ANVÄNDNINGEN

Programsystemet består av:
1) EDITORN för inskrift av källkodsprogram
2) KOMPILATORN som genererar P-koden
3) INTERPRETATORN som gör P-koden exekverbar.

A B C-bladet Samlingsnummer 1980 : 3 sidan 7

Arbetsgången blir: 1) man skapar ett källprogram med editorn, 2) man kompilerar källkoden till maskinkod (P-kod), 3) man testkör programmet med interpretatorn. Om det inte fungerar som man vill, GOTO punkt 1.

Editorn, EDITH är en skärmorienterad editor skriven i Basic främst avsedd att användas för inskrivning och redigering av pascalprogram. Den kan emellertid användas för inskrift och redigering av godtycklig text. Annan editor som lagrar program i sekvensiella filer kan, om man så önskar, användas för att skapa källtexter, t ex EDITOR.

P-koden, som genereras av kompilatorn, är en intern representation av programmet som är optimerad för exekvering. P-koden kan översättas vidare till maskinkod för att snabba upp exekveringen. Ett sådant tillägsprogram är under utarbetande. Metric Mini Pascal går därför i denna version betydligt långsammare än Basic. Om den vidarekompileras kommer den att gå nästan lika fort som maskinkodsprogram.

Programmering i Pascal

Anders Haraldsson

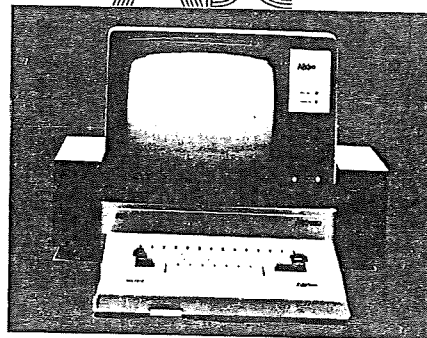


PASCAL

För den som har intresse av programmering i Pascal rekommenderas den utmärkta läroboken Pascal som säljs via Studentlitteratur i Lund.

```
(*MAXNUM READS 4 NUMBERS FROM CONSOLE AND PRINTS THE  
LARGEST. TO EXIT MAKE THE FIRST NUMBER NEGATIVE.*)  
CONST CR=13; LF=10;  
VAR A,B,C,D: INTEGER;  
FUNC MAX4(X1,X2,X3,X4); (*LARGEST OF 4 NUMBERS*)  
      FUNC MAX2(X1,X2); (*LARGEST OF 2 NUMBERS*)  
      BEGIN  
        IF X1>X2 THEN MAX2:=X1  
        ELSE MAX2:=X2  
      END;  
      BEGIN  
        MAX4:=MAX2(MAX2(X1,X2),MAX2(X3,X4))  
      END;  
      BEGIN  
        REPEAT  
          READ (A,B,C,D);  
          WRITE ('THE LARGEST IS ',MAX4(A,B,C,D),CR#,LF#)  
        UNTIL A<0  
      END.
```

Avancerad programmering på ABC80



Anders Isaksson Örjan Kärrgård
METRIC Studentlitteratur

AVANCERAD PROGRAMMERING

Boken Avancerad Programmering på ABC-80 finns nu att tillgå.

Den tar upp många av de tillämpningar och programmeringsmetoder som är nödvändiga för att utnyttja ABC'n effektivt. Den behandlar strukturerad programmering, dataelement, filhantering, assemblerprogrammering och datakommunikation. Programexempel ges liksom praktiska kopplingsexempel för anslutning av yttre utrustning, t ex joystick, annan ABC-80 m m. Vissa programexempel i boken är emellertid inte direkt körbara vilket läsaren bör se upp med genom att ordentligt läsa programlistan.

Datasatser m m behöver tilläggas.

Boken är väl värd sitt pris, ca 105:-

REDOVISNING

ADB-lösningar på redovisningsproblem, likaledes från Studentlitteratur i Lund, rekommenderas varmt för t ex företagare.

Odd Rolander

ABC-80: tal olika mark-RAI Uppsala monterad äldispe iberkort inkt och skortet ABC- ing sk- i kontak- Total rt lösse et. s knapp mycket risk fi som st dock i I övrig ntering ar me 0 kost: 20,63% i övert ekvirer tfn: 011

STYR OCH MÅT MED ABC 80

En ny bok för ABC 80 användare är på väg. Det är STYR OCH MÅT MED ABC 80. Boken har skrivits av Åke Westh, som till yrket är läraren vid Vasagymnasiet i Arboga. Den nya boken ges ut av Studentlitteratur, beräknas bli färdig i oktober och kommer att kosta ca 95 kronor inklusive moms.

Enligt förordet vänder sig boken i första hand till gymnasieskolans tekniskt inriktade linjer, motsvarande arbetsmarknadsutbildningar samt till industrin. För att kunna tillgodogöra sig bokens innehåll fordras kunskaper i digitalteknik, grundläggande datalära och Basic-programmeringen.

Boken baseras delvis på den undervisningsmaterial som Åke Westh använt vid specialkursen i kommunikationselektronik vid Vasagymnasiet i Arboga.

Första delen av boken beskriver V24-snittet och dess användning. De styr- och mättilämpningar som används här är resultat av elevernas projektarbeten. Uppgiften för eleverna har varit att utveckla både hård- och mjukvaran. De framtagna konstruktionerna finns inte att tillgå på den öppna marknaden, påpekar Åke Westh.

Boken är pedagogiskt upplagd. Efter beskrivningen av V24-snittet och presentation av korta programexempel för kontroll av in- och utgångar följer tillämpningarna. Åke börjar med den enkla styrningen av en display för att sedan beskriva ett kort med nio individuella switch-funktioner. Efter det följer ett avsnitt där processkontroll beskrivs. Alla exempel är väldokumenterade med både kopplingsschema, blockdiagram och program i form av såväl listningar som flödesschema.

Den kanske mest intressanta tillämpningen är en logikmonitor. Denna kan, med hjälp av ett anpassningskort testa en IC-krets. På skärmen presenteras då de logiska nivåerna på anslutningarna. Av presentationen framgår också om det på något stift ligger ett pulståg.

I nästa avsnitt beskrivs ABC-bussen och dess expansionsmöjligheter. Minne och utbyggnad av detta berörs också. Sedan följer laborationer med analog/digital, digital/analog och reläkort. Praktiska exempel med ABC 80 som funktionsgenerator och system för komponentprovning med tillhörande program är direkt användbara i praktiken.

Det har nu blivit populärt att med hjälp av en dator samla in och bearbeta mätvärdesinformation. Sista avsnittet i boken berör därför IEC-bussen och dess användning. Här får läsaren veta, att kompletterat med anpassningskortet, kan ABC 80 med fördel användas som mät dator.

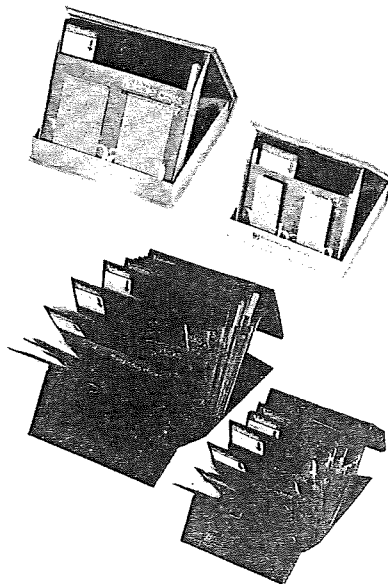
Jag vill avsluta denna presentation med att instämma i Åkes förhoppning om att boken kommer att vara till nytta för både lärare, elever och för dagens industri.

Tad Gruber

Dataton presenterar ett nytt, mikroprocessor baserat projektorstyrningssystem. Systemet kännetecknas av stor flexibilitet och utvecklade programmeringsmöjligheter. Tekniskt sett bygger systemet på att användaren vid varje projektor har en mikrodator som i serieform tar emot och utför instruktioner. Instruktionerna sänds ut från en dator eller finns lagrade på ljudband ihop med ljudinformation som tillhör projektorprogrammet. Som dator används i första hand Datatons egen POLY-VISION COMPUTER 3300. Dock kan vilken annan dator som helst användas med serie utgång och RS-232C snitt. Kommunikationen kan antingen ske i ASCII-kod eller i packad binär form. Upp till 56 projektorer kan kopplas till samma programleverantör. Styrningen till projektorerna kopplas i en lång kedja (daisy-chain) och varje projektor tilldelas en adress med hjälp av tumhjulskomplare. Instruktioner finns för toningar av ljuset från 0 till 99 sekunder, sättning av ljusstyrka från 0 till 99%, stega fram eller bak en bild. Operatören kan också beordra att stega till godtycklig bild i magasinet. Den närmaste vägen väljs då och maximal åtkomsttid blir ca 50 sekunder när Kodaks karusellprojektor används. För ytterligare information om styrsystemet kontakta Dataton AB, tfn 013-16 09 11 och tala med Björn Sandlund eller Mikael Fahl.

FRÅN VERBATIM

BHJ Datautrustningar AB kan nu genom sin leverantör Verbatim, världens ledande tillverkare av disketter, presentera en minidiskett speciellt framtagen och testad för Micropolis-enheter. Denna diskett har 77 spår och dubbel packningstäthet. Till skillnad från standard disketten har Verbatim MD 577: - förstärkningsring i mitten, - är testad 2 ggr (A - standard certifier, B - Micropolis certifier), - magnetskiktet är avsett för dubbel packningstäthet. MD 577 finns med olika sektoriseringar (577-01, 10, 16); BHJ Datautrustningar kan även som första leverantör presentera skrivhjul för QUME och DIABLO med Sverige/Suomi-sekvens. I sortimentet ingår följande sex stilsorter: Orator, OCR-B, Courier 10, Letter Gothic 12, PICA 10, Prestige Elite 12. För övrigt finns i sortimentet ca 80 olika stilsorter. För ytterligare information om produkter från BHJ Datautrustningar ring tfn 08-24 94 80.



AKUSTISKT MODEM

MIKROTEKNIK DATA AB meddelar, att arbetet just avslutats med ett akustiskt kopplat modem. Det har passerat föreskrivna tester och uppfyller alltså Televerkets normer.

Modemet är uppbyggt på ett kretskort i europa-formatet och karakteriseras därför av låg vikt och liten volym vilket gör att det är speciellt lämpat för mobilt bruk. Modemet passar alla typer av terminaler med standard RS232 (V24)-anslutning och överföringshastigheten är upp till 300 baud. Speciellt konstruerad och elegant lurhållare passar dialog telefonen. Med ljudioder i mikrofon Till/Från läge och Carrier-signalen. MODEM 910 inkl batterieliminatorkostar 2 100 kr. För inmontering i egna styren finns MODEM 910/C på europakort och med separat lurhållare. Även den modellen kostar 2 100 kr.

MODEM 910/A med 9-polig kontakt för direkt anslutning till ABC 80 kostar 1 950 kr. För ytterligare information kontakta Mikroteknik Data AB tfn 0758-125 25.

LAGERPROGRAM LETAR LÄTT

Från Datakraft AB aviseras en betydligt förbättrad programprodukt för datorbaserad lagerregistrering. Det nya programmet Lager 80 har på ABC 80 en maximal söktid på tre sekunder. De olika posterna lagras separat och indelas i huvud- och undergrupper. Bläddring i lagret kan ske med tre poster per sekund och sökning av ett specifikt varunummer tar mindre än en sekund. Selektion av varugrupp kan genereras enkelt och med en hastighet av 100 poster per minut. För ytterligare information kontakta Datakraft AB, tfn 046-11 45 80.

NYA DATADIS

DataDisc program för hösten. Förutom skiva kommer alla modeller kompatibla enheter med köra program kapacitet. Me användaren välja täthet respekt lagring för fem på 8-tums IBM-format, användas för system.

Alla nya modeller packningstäthet även användas ABC 80 i och DataBoard 468 platser är av 80's primärminn övriga platser IN/UT-kort för till yttre enhet och färgmonitör.

PRODUKTNAMN (modell)

DATADISC 80
DATADISC 81
DATADISC 82
DATADISC 83
DATADISC 84

DataDisc enhetning via ett i fläktkyllning min vilket ökar liv DataDisc le ning till ABC svensk bruksan av TRANFOR i samarbete Marknadsförin 08/730 57 30.

NY FÖRPACI

En klart fört och mer änd Luxor på för p ABC 80. Den i programkassett dokumentation användas för s



A B C-bladet Samlingsnummer 1980 : 3 sidan 11

MASKINSPRÅK PÅ ABC 80

-KAN DE VA' NÅT ?

Av Per Lindberg - The mad programmer

Den programmerare som verkligen vill utnyttja sin dator till 110% kommer snabbt att få behov av att skriva åtminstone vissa delar av sitt program direkt i maskinspråk. Det kan gälla tillfällen då t ex tidskritiska operationer ska utföras, exempelvis ligga och vänta på någon statussignal och sedan snabbt läsa in data från en yttre enhet. Andra exempel är en musikrutin som lägger ut en frekvens på en datautgång, där en exakt kontroll över frekvens och tid behövs eller en snabb sorterings- eller sökrutin.

På en persondator typ ABC 80 är det att komma åt maskinspråksnivån från Basic. Med POKE, PEEK och CALL kan vi skriva och läsa direkt i minnet, samt anropa maskinspråksrutiner.

Låt oss antaga att vi har kopplat en ABC 80 till en våg, där vi kan läsa in viktigen från ett interface som är kopplat till en I/O-port. Men det är bråttom att läsa in data, för det är flera siffror som kommer snabbt efter varandra. Så vi skriver en liten rutin i Assembler för att sköta den detaljen. Resten av programmet skriver vi naturligtvis i Basic.

När assemblerprogrammet är skrivet, ska det assembleras, dvs översättas från instruktioner typ "LD A,(HL)" och "CALL FOOBAR" till motsvarande maskininstruktioner. Detta kan för det mesta göras för hand. Det är bara att räkna ut eller slå upp varjeinstruktion för sig. Är det däremot ett större program (mer än 10 rader) så är det bra att ha ett program som utför assembleringen - en så kallad Assembler. Men det GÅR att "handassemblera" längre program. Det gäller bara att inte göra några fel.

När så programmet är assemblerat, ska det läggas in byte för byte någonstans i minnet. Nu kan vi göra på flera olika sätt. Större datorsystem har oftast någon form av Loader - ett program som läser från en fil och lägger upp kod i minnet. Ett liknande tillvägagångssätt kan tillämpas på ABC 80, där huvudprogrammet kan läsa från fil och lägga upp rutinen. Detta är särskilt användbart när vi har ont om utrymme i huvudprogrammet. Här ett exempel på hur det hela kan se ut:

```
100 A%=65408% : N%=17% : REM STARTADDRESS
    & ANTAL BYTES SOM SKA LADDAS
110 FN "FOO.D1" ASFILE 1%
120 I%=0% TO N%-1% : INPUT $1%,D% :
    POKE A%+I%,D% : NEXT I%
130 CLOSE 1%
```

En liten nackdel med denna metod är att det blir fler filer att hålla reda på, vilket kan bli ganska jobbigt om det inte finns en flexskiveenhet och filen måste sparas på kassett. En variant är då att lägga maskinspråksprogrammet i DATA-satser. Programmet blir ganska likt föregående exempel:

```
100 A%=65408% : REM STARTADDRESS
110 RESTORE 1000
120 READ N% : REM ANTAL BYTES SOM SKA L
    ADDAS
130 FOR I%=0% TO N%-1% : READ D% : POKE
    A%+I%,D% : NEXT I%
```

```
1000 DATA 17
1010 DATA 33,0,0,219,0,203,103,192,219,3
    2,203,111,200,33,37,252,201
```

Tyvärr går det åt ganska mycket minne för att lagra data på det här viset, i stort sett en byte per tecken. Ett mer minnessnålt lagringssätt är att lägga data direkt i POKE-satser:

```
100 A%=65408% : REM STARTADDRESS
110 POKE A%+0%,33%,0%,0%,219%,0%,203%,10
    3%,192%,219%,32%
120 POKE A%+10%,203%,111%,200%,33%,37%,2
    52%,201%
```

Här lagras varje byte som ska "pokas" in som ett heltal, vilket endast tar upp två byte. Tyvärr blir samtidigt programmet en aning mer svårsläst.

HÄMTNING AV KOMMANDON

När vi nu har lagrat upp en maskinspråksrutin, vill vi anropa den. På ABC 80 finns det två nästan likadana funktioner som kan göra det: CALL () och CALL(.). De är funktioner och ska användas t ex som SQR()-funktionen. Exempel:

```
100 X%=CALL(A9%)
```

Om A9%=65408% så hoppar programmet till adress 65408 och utför maskinspråksprogrammetsom förhoppningsvis ligger där. När vi senare kommer till maskininstruktionen RET (dvs 201 decimalt) återgår vi till vårt Basic-program och variabeln X% får värdet av innehållet i registerpar HL i Z 80-processorn. Ibland har vi ingen glädje av detta, men oftast kan det vara bra att kunna överföra värden från maskinspråksprogrammet. Vi kan även överföra ett värde till maskinspråksprogrammet. Då använder vi en liknande programkonstruktion:

```
100 X%=CALL(A9%,P3%)
```

Här fungerar det precis som i det tidigare exemplet, bara med den skillnaden att värdet av variabeln P3% kommer att läggas i register DE i Z 80-processorn vid anropet. Vi kan alltså överföra ETT heltal in till rutinen, och få tillbaka ETT heltal vid återhoppet. Vill vi överföra fler, måste vi "poka" undan dessa någonstans i minnet. Då är det smidigt att låta den överförda parametern istället utgöra ADRESSEN till var data finns. Sedan är det programmets sak att gräva upp dessa. Tekniken går givetvis att använda i båda riktningarna, både vid inhopp till maskinspråksrutinen och återhopp till Basic-programmet.

LAGRING

Var ska vi lägga upp våra maskinspråksprogram och "poka" undan våra data? Tittar vi på en minneskarta på ABC 80, ser vi att det finns lite ledigt minne högst upp. Men det är bara 128 byte och den arean är lämplig att använda till andra saker, t ex någon slags "COMMON"-area att spara värden på viktiga variabler vid CHAIN av andra Basic-program. Den fiffige kommer även att upptäcka att det finns 64 byte i bildminnet som aldrig används. Men att använda DET är verkligen inte att betrakta som "eleganta programmering"!

Det finns dock två andra sätt att få ledigt minne i ABC 80. Det ena är att flytta "taket" eller "golvet" i den area som normalt används till Basic-program. På adress 65052 och 65053 ligger "BOFA", dvs en pekare till "golvet", och på adress 65063 och 65064 ligger adressen till "taket". Genom att ställa om dessa med POKE kan minne läggas undan. Det enklaste är nog att lyfta "golvet". Det kan göras antingen som ett kommando eller som en sats i ett program. Antag att vi vill sätta "BOFA" till 5000. Då kommer kommandot att se ut så här:

```
POKE 65052,50000,SWAP%(50000) : NEW
```

Och i ett program kan vi göra så här:

```
1000 POKE 65052,50000,SWAP%(50000) : CHA
    IN ''
```

Observera att vi i båda fall måste nollställa Basic med NEW. Satsen CHAIN " " ger samma effekt som kommandot NEW. Vill vi, så går det utmärkt att köra vidare i ett nytt program med exempelvis CHAIN "KAKA2". Den här metoden är dock ganska omständlig - vi måste dela upp Basic-programmet och ibland finns det en printerrutin lagrad under "golvet" och kanske extra minne är inkopplat. I så fall ligger "golvet" från början MYCKET längre ner. Jag brukar föredra att använda metod nummer två: TA UPP MINNE PÅ HEAPEN.

INTERNA ADRESSER

I den area, som finns tilldelad för Basic-program, läggs själva programmet upp som en intern kod från "golvet" och uppåt. Det finns en pekare som visar var det tar slut. Den heter "EOFA" och ligger på adress 65054 och 65055. Den är dock ganska ointressant. Från "taket" och nedåt växer stacken, som är en minnesarea, som används till bl a tillfällig lagring av mellanresultat. Här lagras t ex återhoppadresser till GOSUB-RETURN och återhoppadressen vid CALL. Ovanpå själva Basic-koden ligger HEAPen. Det är en minnesarea som används till att lagra variabler (tal och strängar) när Basic-programmet körs. När vi skriver RUN så skapas heapen innan Basic-programmet går igång. Då reserveras alltså minnesutrymme till alla variabler, dock inte sådana som behöver DIMensioneras. I ABC 80-Basic är det nämligen tillåtet att ha en variabel eller ett uttryck i en DIM-sats och då vet Basic inte i förväg hur mycket minne som går åt. Det görs först då DIM-satsen utförs i programmet.

På adress 65056 och 65057 ligger pekaren "HEAP" som talar om var heapen slutar. (HEAP pekar egentligen på första lediga byte EFTER heapen). Här kan vi lägga upp maskinspråksprogram eller minnesareor för POKE och PEEK. Det är bara att läsa HEAP-pekaren, addera den mängd som önskas och sedan skriva tillbaka "HEAP". Eftersom vi inte i förväg vet VAR vi kommer att få minne, måste vi spara undan det gamla HEAP-värdet i en variabel för att senare hitta dit. Dessutom måste vi se upp så vi inte kör genom "taket"! Vi måste faktiskt ta till en säkerhetsmarginal till stacken också. Ca 500-1000 byte brukar räcka.

Så här kan det gå till att reservera allt kvarvarande minne till en minnesarea som ska användas t ex till en stor bufferarea för en editor:

```
100 S%=PEEK(65063%)+256%*PEEK(65064%) :
    REM TAKET
110 H%=PEEK(65056%)+256%*PEEK(65057%) :
    REM HEAP
120 IF S%-H%<1024% THEN PRINT 'MINNET R
    ÄCKER INTE (SNYFT)' : STOP
130 A%=H% : REM SPARA STARTADRESSEN
140 H%=S%-1024% : REM 1024 BYTES UNDER
    TAKET
150 POKE 65056%,H%,SWAP%(H%) : REM SKRI
    V IN NYA HEAP
160 B%=H%-1% : REM SPARA SLUTADRESSEN
```

Eftersom ett heltal tar upp 2 byte och POKE bara skriver ut en byte, är frågan vilken av dem som skrivs ut? Svar: Den lägsta. Så för att lägga ut båda, använder vi funktionen SWAP%() som kastar om den höga och låga byten i ett heltal.

A B C-bladet Samlingsnummer 1980 : 3 sidan 12

Ett exempel till där vi vill lägga in en maskinspråksrutin på heapen:

```

100 RESTORE 1000 : READ N% : REM ANTAL
    BYTES
110 S%=PEEK(65063%)+256%*PEEK(65064%) :
    REM TAKET
120 H%=PEEK(65056%)+256%*PEEK(65057%) :
    REM HEAP
130 IF S%-H%-N%<1024% THEN PRINT 'MINNE
    T RÄCKER INTE (SORRY)' : STOP
140 A%=H% : REM SPARA STARTADRESS
150 FOR H%=H% TO H%+N% : READ D% : POKE
    H%,D% : NEXT H%
160 POKE 65056%,H%,SWAP%(H%) : REM SKRI
    V IN NYA HEAP
500 REM
600 REM
700 REM
1000 DATA 17
1010 DATA 10,33,<O.S.V>
    
```

Lägg märke till att när vi gått ur slingan på rad 150, kommer H% att vara ett mer än sista värdet på H% i FOR-slingan. Så när vi sparar undan H% på rad 160 kommer "HEAP" att peka på första LEDIGA byte. Den här metoden att lägga upp maskinspråksprogram har dock enallvarlig nackdel - vi vet inte i förväg VAR någonstans den kommer att hamna. Därför måste vi spara undan startadressen i t ex A% för att hitta dit. Det innebär även att vi tyvärr inte kan använda instruktioner av typen "JP(adr)", ty vi vet inte i förväg vart vi ska hoppa. Jump Relative-instruktionen går dock utmärkt att använda, och det räcker gott vid programmering av ABC Z 80-processorn.

INGA PATENTLÖSNINGAR

Som synes finns det inga patenti lösningar för att lägga upp maskinspråksrutiner, utan vi får anpassa metoden efter omständigheterna. De ovan beskrivna metoderna borde dock täcka de allra flesta situationerna på ABC 80. Den som vill lära sig mer om sådana saker rekommenderas t ex att läsa boken "Avancerad programmering på ABC 80" av Anders Isaksson och Örjan Kärrsgård.

Om du tycker om att experimentera, kan du försöka ladda ett maskinspråksprogram direkt i bildminnet!

Till sist ett gott råd: Spara för allt i världen undan programmet på kassett eller skiva INNAN du provkör det! Även om du bara har ändrat någon till synes obetydlig detalj! Det finns nämligen alla chanser att göra fel, och det brukar resultera i att maskinen "spårar ur" och programmet går förlorat.

NYFIKEN

Av Magnus Lundberg, som själv kallar sig Skånela Datorerare, fick vi följande brev:

"Vad finns egentligen på skivan? Och hur skall jag kunna se hur det lagras? Det vare väl kul att ha möjlighet att gå rakt in på skivans spår och se vad där döljer sig. Inte dumt att kunna ta en säkerhetskopiering på skivan heller. Och gärna på band! Sådana här "utility"-program har inte varit populärt att ta fram! Undrar du varför? - Jo, det skulle ju då kunna avslöjas hur de där "smartingarna" gör för att skapa filer omöjliga att lista. Nu är närapå inget omöjligt - bara du kan knepen (som ingen vill lära ut). Därför har jag tagit fram några praktiska program som jag har stor glädje av. Bland annat för att få "run-only" program att bli lite normalare (listbara). Vid mina övningar har jag stött på allehanda fiffiga sätt att sabba för den nyfikne. En skiva kan prepareras ganska rejält, så att den inte uppfattas normalt av de vanliga programmen. Libprogrammet säger "discfel" etc, men skivan går ändå att använda normalt! Har du tänkt på att programmakarna brukar ha en fast programmerad "meny" i sina system? Varför

inte ett generellt menyprogram? Nedan visas ett praktiskt programexempel. De skivor som sitter i kommer att läsas av programmet och alla körbara program visas på skärmen med en siffra framför. I rubriken ser du vilka skivor som sitter monterade. Det är meningen att du endast ska behöva slå in den siffra som står framför det program du vill köra, så läses detta in automatiskt och börjar exekveras. Vill du bygga ett system med denna teknik skall du se till att dina program avslutas med ett kommando "chain" så menyn åter plockas in."

Så långt Magnus brev. Intresserade som vill veta mer kan kontakta honom per tfn 0760-380 25.

Här nedan ser du Magnus program Meny.

```

10 REM $ list meny Ver 80-08-09 *
20 REM ** (c) Magnus Lundberg *
30 REM ** Nyfors *
40 REM ** 190 40 Rosersberg *
50 REM ** tel 0760-380 25 *
60 REM ** 1980-06-06 *
70 L%=0% : OPEN 'PR:' ASFILE L%
80 E2%=12% : REM Längd per pgm
90 E$='BAC' : REM Extension att välja
100 DIM P$(99%)=16%,P1$(1%)=12%
110 REM $ Ange vilken drive
120 FOR D%=0% TO 1%
130 REM $ Läs indexet
140 T%=0% : S%=6% : GOSUB 350 : IF Z%<>0
    % 290
150 A%=62959% : FOR I%=0% TO 7% : V%(I%)
    =PEEK(A%+I%) : NEXT I%
160 REM $ V-vektorn innehåller nu
170 REM $ de sektorer som har VTOC
180 REM $ Sök igeom dessa efter 'EXT'
190 T%=2% : FOR S%=0% TO 7%
200 IF V%(S%)>1% GOSUB 350 ELSE 280
210 IF Z% : 'FELFEL'Z% : STOP
220 A%=62720%+16% : REM Start på dosbuf
    0
230 FOR N%=0% TO 15% : GOSUB 370
240 REM $ Skivans lbl
250 IF INSTR(1%,N$,'LBL') P1$(D%)=LEFT$(
    N$,8%) : P1$(D%)=LEFT$(P1$(D%),INSTR
    (1%,P1$(D%)+', ')-1%)
260 IF INSTR(8%,N$,E$) P$(P%)=LEFT$(N$,8
    %) : P%=P%+1%
270 NEXT N%
280 NEXT S%
290 NEXT D%
300 REM $ Programnamnen nu i P$(i)
310 GOSUB 390
320 REM Logga
330 ; CUR(20,0); : CHAIN P$(I%)+'.'+E$
340 REM $ Sub för läsn av disc
350 POKE 64769%,D% : Z%=CALL(24678%,SWAP
    %(T%)+S%*32%) : Z%=PEEK(64789%) : RE
    TURN
360 REM $ Läs ut filnamnen
370 N$=' ' : FOR I%=4% TO 14% : N%=N$+CHR
    $(PEEK(A%+I%)) : NEXT I% : A%=A%+16%
    : RETURN
380 REM $ Menyn visas med val
390 ; $L%CHR$(12%);
400 P%=P1$(0%)+P1$(1%) : IF LEN(P%)>8% P
    %=P1$(0%)+' & '+P1$(1%)
410 ; $L%"ProgramLista("E$") : "P$
420 ; $L%STRING$(40%,61%)
430 FOR R%=0% TO 18% : FOR K%=0% TO P% S
    TEP 19% : I%=R%+K%
440 I$=RIGHT$(NUM$(I%),2%)
450 IF I%<10% I$=' '+I$
460 IF LEN(P$(I%))<E2% P$(I%)=P$(I%)+ '
    : GOTO 460
470 IF P$(I%)=SPACE$(E2%) 490
480 ; $L$I$ 'P$(I%);
490 NEXT K% : ; $L% : NEXT R%
500 ; CUR(22%,28%)'Nr: ';
510 INPUTLINE $L%A$ : A$=LEFT$(A$,LEN(A$
    )-2%) : ONERRORGOTO 500
520 I%=VAL(A$) : IF I%-P%+1% 500
530 IF I%<0% 500
540 RETURN
    
```

Så här ser skärmen ut när du kör programmet med MENU och skivan i DR0 heter UTILITY och den i DR1 heter TEST.

Programlista(BAC): Utility & Test

0	UTILITY	19	RENSA
1	MENY	20	TEXT85
2	HELP	21	JUST85
3	LIB	22	JUST130
4	EXTRACT	23	SKRIV
5	AVBRYT		
6	BEHÖR		
7	BASICODE		
8	DISKSTAT		
9	VARIABEL		
10	MENY		
11	T&O		
12	MARG		
13	PRINTER		
14	DAGSTART		
15	JUST		
16	PLOT		
17	GRAF		
18	FORM		

Nr :

PAUSEDEM

PAUSEDEM är på bildskärmen 8240 kan göras om du t ex därefter i lu från det ställ Programmet fungerande pr Programmet ända till des tangenten. C väntar på att kas ned. Programmet exempel på " 10 står REM därvid mycl under namne: editera rad IC SAVE, backa ordet REM o fram cursorr och därefter att radnumre endast står ! om cursorn, RETURN, up kommando a 10 kommer som program vissa delar Denna teknik utvecklar pr flera versioner att ta bort programmet t Säkert har d någon enstak gram.

PROGRAMLISTA

10	REM SA
	(C) OL
20	REM \$:
	ör tel
30	REM \$:
	rift e.
	rutin
40	REM * 1
50	REM * '
60	REM \$ 1
70	REM
80	GOTO 11
90	REM # :
100	IF INP
110	; CHR\$(
	.CHR\$(
120	POKE 61
	HO\$)<>
130	; CHR\$(
	G';
140	RETURN
150	REM # :
160	; CHR\$(
	; STR
	nktion
170	; 'Ate
	RING\$(
180	; CUR(
	iera
190	FOR I%
200	; CUR(
)
210	; CUR(
220	A\$=INP
	POKE
230	NEXT J

PAUSDEM

PAUSDEM är ett demo-program som visar på bildskärmen hur du med subrutinen 8000-8240 kan göra paus med utskrift på printern om du t ex får ett telefonsamtal, för att därefter i lugn och ro fortsätta utskriften från det ställe där du stoppade.

Programmet PAUS är ett exempel på ett fungerande program där subrutinen lagts in. Programmet skriver ut texten på rad 3 ända till dess du trycker ned mellanslagstangenten. Då slutar printern skriva och väntar på att RETURN-tangenten skall tryckas ned.

Programmet PAUSDEM visar även ett exempel på "smart" programmering. På rad 10 står REM SAVE SKRVPAUS. Du kan därvid mycket enkelt spara programmet under namnet SKRVPAUS.BAC genom att edit rad 10, köra fram cursorn till ordet SA backa cursorn så att du tar bort ordet REM och radnumret 10, på nytt köra fram cursorn till efter ordet SKRVPAUS och därefter trycka RETURN. I och med att radnumret tagits bort och på skärmen endast står SAVE SKRVPAUS, till vänster om cursorn, i det ögonblick du trycker RETURN, uppfattar ABC-80 detta som ett kommando att spara programmet. Rad nr 10 kommer dock att sparas i sin helhet som programrad, trots att du tagit bort vissa delar av raden genom editeringen. Denna teknik är mycket användbar då man utvecklar program. Särskilt om man har flera versioner med liknande namn. Risken att ta bort fel version genom att spara programmet under fel namn kan då minskas. Säkert har du råkat ut för att inte bara någon enstaka gång ha tagit bort fel program.

PROGRAMLISTA PAUSDEM.BAS

```

10 REM SAVE SKRVPAUS,1.0,1979-07-15,
(C) Olle Mauritzon
20 REM $ Stoppa pågående utskrift (f
ör telefonsamtal)
30 REM $ Anropas omedelbart före utsk
rift eller inlägges i utskriftssub
rutin
40 EM # Numreras 8230,2
50 EM # VARIABLER
60 REM $ H$ = hjälpvar
70 REM
80 GOTO 150 : REM Demoprogram
90 REM # SKRIVARPAUS
100 IF INP(56%)<>160% 140
110 CHR$(13%)'ATERSTARTA MED RETURN
CHR$(8%);
120 POKE 65013,0% : GET HO$ : IF ASC(
HO$)<>13% 120
130 ; CHR$(13%)'GÖR PAUS MED MELLANSLA
G';
140 RETURN
150 REM # SKRVPAUS-DEMO
160 ; CHR$(12) '* SKRIVARPAUS-DEMO * ' :
; STRING$(40,61); 'Demonstrerar fu
nktionen utan att förstöra paper.
170 ; 'Åter till MENY med "e" . ' : ; ST
RING$(40,45);
180 ; CUR(8,0); : GOSUB 130 : REM Init
iera
190 FOR I%=12 TO 21 : FOR J%=0 TO 39
200 ; CUR(I%,J%)CHR$(RND*(127-32)+32,7
)
210 ; CUR(8,0); : GOSUB 90
220 A%=INP(56)-128 : IF A%=64 OR A%=96
POKE 65013,0 : CHAIN 'MENY'
230 NEXT J% : NEXT I% : GOTO 190

```

PROGRAMLISTA PAUS.BAS

```

1 OPEN "PR:" ASFILE 1
2 GOSUB 8000
3 ; #1,"HEJ TELEFONEN RINGER"
4 GOTO 2
8000 REM
8230 REM SAVE SKRVPAUS,1.0,1979-07-15,
(C) Olle Mauritzon
8232 IF INP(56%)<>160% 8240
8234 ; CHR$(13%)'ATERSTARTA MED RETURN
CHR$(8%);
8236 POKE 65013,0% : GET HO$ : IF ASC(
HO$)<>13% 8236
8238 ; CHR$(13%)'GÖR PAUS MED MELLANSLA
G';
8240 RETURN

```

* READY

KNAPPSATS ATT BYGGA

Av Lars Gunnar Karlsson (SM4FVD) i Hagfors fick vi några rader tillsammans med beskrivning av en knappsats och ett kort program.

Den oinvidige kan här få en inblick i vad som kan göras med en dator och dess V24-kontakt, eller som Lars Gunnar skriver: "...Många vill nog kunna ansluta någon enkel grej till datorn för att styra en del komandon. Nedan beskriven låda har fem knappar. Dessa kan exempelvis styra markören (cursorn) i alla riktningar eller starta och stoppa en eller flera "klockor". Knappsatsen kan även användas tillsammans med olika spel som paddel, sänka båtar, rita på skärmen osv. "5-knapp's programmet" måste naturligtvis läggas i ett huvudprogram, men det är ju inte så svårt! Jag har även gjort en variant av programmet där datorn känner av fönster och dörrar och talar om vilka som är öppna."

Hälsningar
"Lasse Data"

```

10 REM ** LASSE DATA 23-05-1980 **
20 REM ** LASSE KARLSSON HAGFORS **
30 REM ** PRAKTISK ANVÄNDNING AV **
40 REM ** V-24 INGÅNGEN **
50 REM ** V-24 INGÅNGEN **
60 ; CHR$(12%)
70 A=INP(58) : C=128 : FOR I=2 TO 0 STE
P -1
80 IF A AND C THEN B(I)=1 ELSE B(I)=0
90 C=C/2 : NEXT I : U=A-130
100 ; CUR(2,0)'DATORN LÄSER AV BITARNA 0
,1 OCH 2 PA'
110 ; CUR(4,0)'INPORT 58. (V-24 KONTAKT
EN)'
120 ; CUR(20,25)' LASSE DATA'
130 IF U=5 ; CUR(12,10)'INGEN KNAPP NED-
TRYCKT '
140 IF U=0 ; CUR(12,10)'KNAPP NR 0 ÄR NE
D-TRYCKT '
150 IF U=-2 ; CUR(12,10)'NÅGON TRYCKER P
Å 2 KNAPPAR '
160 IF U=1 ; CUR(12,10)'KNAPP NR 1 ÄR NE
D-TRYCKT '
170 IF U=2 ; CUR(12,10)'KNAPP NR 2 ÄR NE
D-TRYCKT '
180 IF U=3 ; CUR(12,10)'KNAPP NR 3 ÄR NE
D-TRYCKT '
190 IF U=4 ; CUR(12,10)'KNAPP NR 4 ÄR NE
D-TRYCKT '
200 GOTO 70

```

BREV

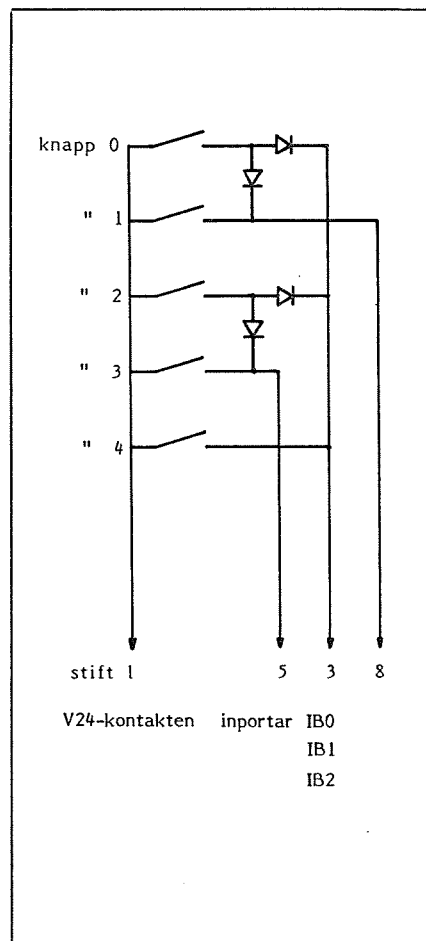
Omvandlar din printer till en elektrisk skrivmaskin men saknar ordbehandling. Programmet kan endast rad för rad skriva ut det du matat in.

PROGRAMLISTA BREV.BAS

```

10 ; CHR$(12)
20 REM *** BREV ***
30 REM *** H.H. ***
40 ; CUR(5,5)'SKRIV DITT BREV PÅ PRIN
TERN"
50 ; CUR(7,5)'NÄR RADEN ÄR FÄRDIG,TRY
CK PÅ RETURN"
60 ; CUR(9,5)'TRYCK PÅ MELLANSLAG SÅ
KAN DU BÖRJA"
70 ; CUR(11,5)'VILL DU SLUTA SKRIV ""
STOP""
80 GET Ø$
90 ; CHR$(12)
100 OPEN "PR:" ASFILE 10'
110 A$=""
120 INPUTLINE A$
130 A$=LEFT$(A$,LEN(A$)-2)
140 IF A$="" THEN 150
150 ; #10A$
160 ;
170 GOTO 110
180 CLOSE 10
190 END

```



5 st återfjädrande tryckknappar
4 st dioder (IN4148)
kabel (4 ledare)
V24 kontakt
en låda
På V24-kontakten används inportarna IB0, IB1, IB2
+9V släpps till inportarna. Maximalt kan den presenterade lösningen erbjuda 6 olika kombinationer.

FÖRSLAG OM STARTANDE AV ETT ABC-FÖRLAG

Detta intressanta inlägg har kommit från Kjell-Åke Johansson

Förlagsverksamhet är något som ABC-klubben borde bedriva. Vid sidan om utgivning av ABC-bladet kan man tänka sig årsböcker och naturligtvis program.

Att distribuera program i form av listningar är, vare sig det sker i böcker eller i ABC-bladet, ganska opraktiskt för användaren. Listningen kan ha ett värde för den som saknar printer, men att knappa in ett program är ganska mödosamt och ofta introduceras svåra fel. Eftersom de program som finns i ABC-bladet en gång har funnits på ett annat mera lätthanterligt medium, kassetter eller flexskiva, känns det mindre gsfullt att skriva av programlistningarna.

En lösning på detta problem och lämplig start för ett ABC-förlag vore en utgivning av de program som förekommer i ABC-bladet på ett annat medium.

Eftersom det finns flera olika flexskivemedier är kassetten lösningen. Kassetten är ett medium som binder alla ABC 80-användare samman. Även program som skall köras på floppy kan distribueras på kassetter.

Kostnaden för massproduktion av kassettkopior är mycket blygsam. Enligt ett företag jag haft kontakt med skulle den bli ca 5 kronor för C 60 och en krona mer för C 90, och då får man kopiering på båda spåren. Jag har testat sådana kopior och funnit att de fungerar.

Det här betyder att man kan låta medlemmarna prenumerera på kassettkopior av de program som förekommer i ABC-bladet. Kostnaden för fyra kassetter per år skulle inte behöva bli större än 35-40 kr. Kopieringstekniken kan utnyttjas också för annan utgivning av program - här kommer i punktform några förslag på verksamheter för ett ABC-förlag:

1. Kassettbilaga till ABC-bladet.

2. Utgivning av program. Medlemmar och andra skulle kunna ge ut program genom ABC-förlaget. Kopieringskostnaden är ca 5 kr, så att om programmen säljs för ca 40 kr kan författaren få 15-20 kr per såld kassetter. Genom att hålla låga priser minskar man intresset för kopiering i lönn. En redaktion som undersöker inkomna förslag behövs naturligtvis - en chans för många duktiga ADB-studerande att debutera som programmerare?

3. Månadens program - en motsvarighet till bokklubbarnas Månadens bok. ABC-förlaget bjuder in proffs att skriva program, som sedan erbjuds de medlemmar som gått in i Månadens program. Med ca 500 medlemmar och en medlemsavgift på 300 kr får man en budget på ca 150.000 kr.

4. Andra aktiviteter för att stimulera programskrivandet. Ex vis tävlingar - vem skriver det bästa Adventure-programmet, vem skriver den bästa Assemblereditorn med syntaxkontroll? etc. Priser i form av tillbehör till ABC 80.

Allt syftar naturligtvis till att förändra den orimliga situationen som råder för den hårt prövade amatören. För den som inte kan finansiera sin ABC 80 genom arbetsgivare eller som inte är det själv, arbetsgivare alltså, gör denna hobby irreparabla hål i hushållskassan. En jämförelse med andra

käpphästar kan vara på sin plats: en grammo-fonskiva kostar ca 45 kr, en pocketbok ca 50, men ett program till ABC 80 kostar ofta 600 kr. Vem har råd med det? Målet bör vara att en programkassetter inte skall kosta mer än 100 kr och oftast mindre än det!

Några ord kring insända program

Det är med stor glädje som vi på ABC-bladet emottager olika insända programbidrag från er läsare. Ett problem har dock uppstått i samband med den här verksamheten. Ibland är upphovsrätten till programvaran inte helt klarlagd. I fortsättningen kommer ABC-bladet därför att införa en obligatorisk uppbyggnad av programmens början, så att det här klart framgår såväl programnamn som copyrightinnehavare. Vidare vill vi ha uppgift om vem som sänt in programmet o s v.

Totalt sex punkter som enligt vårt förslag ska ha en utformning som i exemplet nedan.

```
10 REM *****
20 REM * TIDEN.BAC *
30 REM * VER 1.0 / 1980-12-02 *
40 REM * (C) ----- *
50 REM * gjort av Kaj Arnö *
60 REM * mod av Bengt Olwig *
70 REM * insänt av ----- *
80 REM *****
```

Genom att som ovan ordentligt ange olika programuppgifter är det enkelt att spåra källan till ett program. Om det dessutom finns en uppgift om adress eller telefonnummer till den person som sänt in programmet kan andra läsare ges tillfälle att kommentera och diskutera programlösningen med insändaren.

Vi tror att en sådan verksamhet verkar stimulerande på programutbytet. Samtidigt är det vår förhoppning att olika programs 'stamträd' genom den obligatoriska märkningen ska framgå på ett sätt som förhindrar oavsiktliga copyright-intrång.

När det gäller insända program är det också önskvärt med REM-satser som med lämpliga intervall ger information om programstruktur och speciella variabler som används i programmet. Denna typ av information tar förvisso en hel del minnesutrymme, men samtidigt underlättar REM-satserna för läsaren att förstå programmakarens intentioner och programmets funktion.

Smartkodning med invävda programstrukturer och en total avsaknad av REM-satser kan vid en första anblick imponera, men sett i ett längre perspektiv är sådan programmeringsteknik enbart av ondo. Vi på ABC-bladet vill således i fortsättningen gärna se väldokumenterade och 'raka' program där finessen ligger i att alla läsare har en god chans att följa med i programmerarens tankegångar. Men låt nu inte våra förmaningar vara en hämsko. Så är inte syftet. Däremot får vi nog alla väl av att dokumentera vad vi gör, så varför inte försöka redan nästa gång som du ska göra ett program...

MJUKVARA, HÅRDVARA OCH FRI KONKURRENS

Det börjar nu finnas allt mer kringutrustning att tillgå till ABC 80. När det gäller flexskiveenheter finns det för närvarande åtminstone fyra olika fabrikat. Dessa är alla lite olika och kräver därför att programvaran läggs upp på skivan för vart och ett av fabrikanterna. Nu har emellertid utvecklingen gått därefter att en av leverantörerna av flexskiveenheter också har lagt under sig den huvudsakliga delen av programvaruförsörjningen till ABC 80 och inte drar sig för att utnyttja detta i konkurrensbegränsande syfte. Genom att inte tillhandahålla programvara till vissa enheter försöker man göra det ointressant för ABC 80-användarna att köpa dessa enheter.

Det här är naturligtvis något som ABC-klubben måste bekämpa med kraft. Det bästa vore om produktionen och distributionen av programvara sköttes av ett eller flera separata programbolag, som inte hade några intressen på hårdvarusidan och som var skyldiga att tillhandahålla programvara som kan användas på samtliga flexskiveenheter till ABC 80.

Jag hoppas att den här saken kan lösas på något sätt utan NO, men frågan är av principiellt intresse och borde kanske sändas vidare till någon av de utredningar som nu synar datoriseringens konsekvenser. Det behövs kanske en lagstiftning när det gäller programvaran, en lagstiftning som inte ensidigt gynnar leverantörer och producenter.

Det är exempelvis inte självklart att programvara alltid skall säljas, det borde i en framtid vara möjligt att låna program, t ex på biblioteken.

Det hela har med informationsfrihet att göra: program till smådatorer kan i vissa avseenden jämföras med tidningar och böcker. På samma sätt som den som ger ut tidningen eller boken får räkna med att ta en risk måste den som skriver kommersiell programvara ta en ekonomisk risk. Idag har vi en bristsituation när det gäller bra och billig programvara till smådatorer och vi måste vara vaksamma så att denna bristsituation inte utnyttjas av producenterna för att permanenta oacceptabla förhållanden på detta område.

Om man försöker tänka sig in i en framtid med betydligt större tillgång på programvara till rimligt pris så uppstår många frågor. Kanske behövs något motsvarande tryckfrihetsförordningen? Borde inte någon myndighet ha skyldighet att spara program på samma sätt som Kungliga biblioteket har skyldighet att spara böcker och trycksaker till eftervärlden?

Kjell-Åke Johansson

Detta är Kjell-Åkes brev. Vid vår kontakt med honom uppgav Kjell-Åke att det vår svårigheten med att få Metric's Assembler på en Micropolliskiva som fick honom att skriva brevet. Nedanför följer en kommentar från Benny Wahluquist, divisionsschef för Mikrodatordivisionen på Scandia Metric:

"Det är omöjligt för oss att ha alla versioner av mjukvaran, som skulle kunna köras på olika versioner av skiveenheter - vi har inte resurser att testa och garantera funktionsdugligheten.

Vi har inte heller haft några förfrågningar från någon leverantör, som skulle vara villig att anpassa program till sina egna skiveenheter. Men skulle vi få en seriös förfrågan från en distributör, då kan vi tänka oss ett avtal om distribution av våra program."

ABC 80's KORTSERIE FRÅN METRIC

Scandia Metric kan nu erbjuda en komplett kortserie, som kan göra ABC 80 till en perfekt styr- och mät dator.

Det är mätdatorgruppen som sedan flera år tillbaka bygger mätdatorsystem för industrin. Gruppens erfarenhet har resulterat i den aktuella kortserien.

I serien ingår kort för serie- och parallellkommunikation, digital till analog och analog till digital omvandling och anpassning till IEEE-bussen. Till alla kort finns utförliga beskrivningar, som visar kortens uppbyggnad och exempel på användning.

Korten passar i Metric's och Luxor's expansionsenheter.

ADC32P är ett anpassningskort för analoga signaler in till ABC 80. Information från 32 loga ingångar omvandlas till digital data. Kortet kan användas till mätningar med mätinstrument med analog utgång, flödesmätare, temperaturgivare, tryckgivare, tråd-
töjningsgivare osv.

DAC är ett kort som kan användas för styrning av analoga signaler med hjälp av AB 80. Exempel av användning är styrning av process/reglersystem, styrning av externa instrument (ex vis funktionsgenerator), styrning av XY-skrivare eller analog presentation av mätvärden på instrument (ex vis oscilloskop).

PIO, med parallella TTL-signaler, kan användas i t ex digitala mätinstrument och vägar med parallell BCD upp till åtta siffror, statuskontroll av signalnivåer (ex vis 32 larmsignaler), styrning av digital utrustning (ex vis multiplexer eller stegmotorer), styrning av digitala nivåer (ex vis reläfunktioner).

SIO är ett anpassningskort för seriell kommunikation och dataöverföring. Kan användas för anslutning till ABC 80 av skrivare med in/ut, bildskärmsterminaler, dataöverföring mellan flera ABC 80 eller synkron eller asynkron dataöverföring mellan ABC 80 och stordator.

MIO, som i likhet med PIO har ett antal TTL signalgrupper, har även plats för 2k P. Kortet kan användas för skrivare med parallellformat, digitala mätinstrument med parallell BCD (max 4 siffror/kort), remsläsare och remsstans, statuskontroll av t ex 16 larmsignaler eller styrning av t ex 16 reläfunktioner.

IF är ett nytt anpassningskort till den gamla instrumentbussen GPIB, som definieras i IEEE 488,1978. Kortet används för en flexibel anslutning av olika instrument såsom digitala multimetrar, funktionsgeneratorer, skrivare osv med möjligheten att lätt kunna koppla in flera instrument eller ändra på instrumentens uppgift i systemet. IEEE-kortet gör att ABC 80 kan fungera som sändare, mottagare och kontrollenhet på GPIB och därmed manövrera alla funktioner på GPIB.

Följande nya kort är i produktionen och kommer inom kort: RELÄ, OPTO, TRANSISTOR och CTC (counter/timer).

Komplett information och prisuppgifter kan fås från Scandia Metric AB, tfn 08-82 04 00.

SMARTAID

OWOCO AB lanserar ett nytt programmeringshjälpmedel för ABC 80 under namnet SMARTAID.

SMARTAID är en mycket god hjälp för såväl nybörjare som avancerade programmerare. Följande kan göras med hjälp av SMARTAID:

Om man skriver en för lång rad eller glömt något som gör att man får "error" (fel), kan man normalt endast skriva om hela raden. Med SMARTAID's bildskärmseditor kan man rädda denna rad utan att skriva om den.

Man kan med SMARTAID's bildskärmseditor även "packa" flera korta rader till en rad och även tvärtom förstås, dela av en för lång rad.

Skärmen kan rensas snabbt och enkelt.

I stället för att skriva in radnummer efter radnummer kan man med hjälp av SMARTAID få automatisk radnumrering och även omvänt, i stället för att behöva ta bort en rad i taget kan man ta bort hur många rader man vill.

Med hjälp av SMARTAID kan alla variabler i ett program listas.

För att inte uppta något av ABC 80's arbetsminne ligger SMARTAID lagrat i ett PROM på 1k byte och för att inte begränsa inkoppling av andra yttre enheter till ABC 80's busskontakt är SMARTAID försedd med en ingående- och en utgående kontakt.

SMARTAID är inbyggd i en modul som är mindre än 100x70mm.

SMARTAID levereras med en utförlig manual och kostar 695,-inkl moms.

För närmare information kontakta OWOCO AB, tfn 08-94 01 80.

AUTOSTART FÖR ABC 80

En autostart-rutin finns nu för de ABC 80-användare som har tillgång till en flexskiveenhet.

Autostart innebär att maskinen vid spänningstillslag eller reset automatiskt startar upp ett användardefinierat program. Funktionen blir då följande:

Först sker den normala initieringen av maskinen, varefter ABC 80 börjar leta efter ett program med namnet AUTOSTART.SYS som sedan startas upp. Initieringen fortsätter med att leta efter den normala BASICERR.SYS.

När detta är gjort sker den egentliga autostarten genom att användarens Basic-program startas. Det programmet måste vara döpt till START. Om programmet START ej kan påträffas erhålls ERR 21.

Autostart installeras genom byte av det befintliga DOS-prommet mot ett ny. Med det nya prommet följer en skiva och bruksanvisningen.

I system med dubbel packningstäthet får DOS-programmet inte plats. Där placeras det på ett separat sk FIO-kort. Obs att ett DOS för dubbel packningstäthet går ej att använda på en kontroll för enkel packningstäthet och vice versa.

Priset för Autostarten är 850 kronor plus moms.

Ytterligare information kan fås av Scandia Metric AB; tfn 08-82 04 00.

*

Även innehavare av DataDisc80 kan glädjas åt en Autostart-rutin. Sattco's Autostart består av två PROM och en systemskiva, som hjälper användaren att autostart-generera sina egna programskivor.

Vilket program som ska startas när spänningen slås på eller reset trycks in är enbart beroende på vilken programskiva som är insatt i flexskiveenheten.

Ytterligare information om Autostart för system utrustade med DataDisc80 kan fås från Sattco AB, tfn 08-730 57 30.

NYA MODELLER AV FLEXSKIVEMINNE

För alla ABC 80-användare, som har behov av ett större externt lagringsmedia, presenteras nu nya modeller av flexskiveminna FD2 och FD2U.

Genom att tillämpa en annan lagringsteknik (dubbel packningstäthet) har kapaciteten ökat från 2x72k byte till 2x150k byte i de nya modellerna, som heter FD2-D och FD2U-D. Dessa har drivenheter för två flexskivor. En av drivenheterna kan kopplas om så att data och program avsedda för FD2 och FD2U kan överföras till de nya skivorna.

Liksom i de tidigare flexskiveminna är även de nya försedda med strömförsörjning och DOS lagrat i ROM. För expansion av ABC 80-systemet med ut/in kort eller minneskort finns det i FD2-D plats för ett kort och i FD2U-D plats för fyra extra kort.

De som sedan tidigare har FD2 eller FD2U kan komplettera dessa med hjälp av en ombyggnadsatts så att den högre lagringskapaciteten uppnås.

Den i FD2-D och FD2U-D befintliga kontrollerelektroniken innehåller även en printerrutin i PROM för styrning av en printer med Centronics-snitt.

För ytterligare information kontakta Scandia Metric, tfn 08-82 04 00.



NY ASSEMBLER FÖR ABC 80

ASMEDIT FÖR ABC 80

Programutvecklingshjälpmedlet ASMEDIT, som tidigare funnits för olika Intel-system, har nu anpassats för ABC 80.

ASMEDIT för ABC 80 är både editor, assembler och debugger i ett enda program. Den skiljer sig emellertid från konventionella program av denna typ genom att den medger interaktiv programutveckling. Dessutom assemblerar ASMEDIT flera gånger snabbare än någon annan känd assembler.

ASMEDIT har framtagits speciellt för att effektivisera programutveckling för mikro-datorer. Genom sina egenskaper är den emellertid också ett utmärkt läromedel för mikrodatorkurser och självstudier. Till skillnad från andra programutvecklingshjälpmedel är ASMEDIT effektiv och smidig att använda även om bandspelare är enda yttre minnet.

ASMEDIT erbjuder ett enkelt sätt att snabba upp eller förkorta Basic-program med hjälp av inlagda assemblyspråkrutiner. Den är optimalt anpassad till ABC 80's interna strukturer och Basic. Fullständig filhantering på kassett och flexskiva ingår.

ASMEDIT är gjord för användning av Intel's assemblyspråk för processorerna 8080 och 8085, vilka utgör industristandard. Z80-processorn i ABC 80 exekverar också den kod som genereras. En ABC 80 med ASMEDIT kan även användas för utveckling av program till andra 8080-, 8085- eller Z80-baserade system.

Programmet levereras på kassetband eller flexskiva. Utförlig användarmanual på svenska med programmeringsexempel medföljer. Priset är 575:-exkl moms.

För ytterligare information kontakta SYNTHESSE, Klostergatan 8B, 222 22 Lund, tfn 046-14 80 11.



NYTT ORDBEHANDLINGSPROGRAM FÖR ABC 80

Ett nytt ordbehandlingsprogram finns nu tillgängligt för ABC 80-användare. Programmet, som heter ABCORD, är speciellt lämpligt för de som inte kontinuerligt arbetar med utskrift, då det innehåller få styrkommandon och dessutom är enkla att komma ihåg.

Texten kan skrivas in så fort programmet har startats. Alla kommandon visas hela tiden på skärmen. För att förenkla rutinerna har programmet automatisk avstävning. Detta innebär att texten kan skrivas in löpande och programmet själv tar hand om avstävningen. Utskriftsformatet kan väljas oberoende av formatet på den inskrivna texten.

Det interaktiva sättet att arbeta gör att det är enkelt att göra korrekturen i texten är mycket enkla då hela raden kan flyttas bort eller skjutas in. Hela meningar kan skjutas in var som helst, resten av raden flyttas då åt höger av ABCORD.

Enkelt upplagda menyer gör att även nybörjare kan använda programmet. Arbetet kan förenklas ytterligare då inställningar för textutskrift kan sparas för senare användning. Där kan spaltbredden bestämmas, placering av texten och sidnumren och om texten ska ha rak höger marginal eller ej.

En enkel och lätthanterlig manual följer med ABCORD. Ytterligare information om programmet kan fås av Luxor AB, Division Datorer, tfn 0141-16 200 eller Scandia Metric AB, tfn 08-82 04 00.

Metric's programsortiment har utökats med ett nytt assemblerpaket för professionell programmering på ABC 80.

Assemblern, som följer Zilog's mnemonics-standard, är helt flexskiveorienterad vilket tillåter mycket stora program.

Paketet innehåller en mängd hjälpprogram som tillsammans med en väl genomarbetad, användarvänlig bruksanvisning bildar ett kraftfullt instrument för eget programskapande.

I paketet finns bl a:

Skärmorienterad editor

I denna kan man, genom att flytta pekaren godtyckligt i bildrutans text, redigera och nyskapa program (även andra texter) med ett flertal kommandon.

Olika alternativ erbjuds för inladdning av egna program i ABC 80. Man kan t ex ladda maskinspråksprogram direkt i arbetsminnet. Det är även möjligt att omforma program så att de kan anropas som egna kommandon i operativsystemet.

I paketet ingår även ett monitorprogram för provkörning och felsökning av egna program. Där finns möjlighet att stega sig genom programmet (single step) och samtidigt granska och ändra i programmet eller processorns register.

En god hjälp ges också av en disassembler som översätter programmet från maskinspråk tillbaka till mnemonics.

Priset för ABC-ASM, inklusive all dokumentation och programvara, är 800:- exkl moms.

För ytterligare information kontakta Scandia Metric, tfn 08-82 04 00.

NYTT REGISTERPROGRAM FÖR ABC 80

Den fristående programvaruleverantören för ABC 80, ABC DATA, presenterar i sin nyutkomna katalog ett nytt generellt registerhanteringssystem, DBAS, till lågt pris. Programsystemet, som kräver tillgång till flexskivesystem, ger användaren möjlighet att lägga upp alla olika typer av register.

Dessutom finns kraftfulla rutiner för sortering, sökning och uppdatering av register. Sortering görs snabbt med Quicksortrutin. Sökning kan göras i fem olika mod, bl a snabbsökning och sökning av visst ord i hela registret eller viss delpost.

Utskrifter på printer kan göras i valfritt format.

Programsystemet kostar 595 kronor inkl moms.

ABC DATA presenterar samtidigt ett enklare registerprogram, DATREG, för lagring på kassett. Pris: 75 kr.

Ytterligare information om dessa program finns i ABC DATAs katalog som gratis kan rekvireras från ABC DATA, Box 2002, 175 02 Järfälla

Med anledning av ett något oklart läge om vem som marknadsför och säljer EPSON-printer fick vi följande brev från SCANTELE:

...Den internationella marknadsföringen av EPSON-produkter har skett dels via Epson USA, dels via Epson Japan och i Europa via agentnät med stöd av OEM-försäljning från Tyskland.

Från den 1.8.1980 har baserat på erfarenhet försäljningen helt förts över till respektive generalagent med teknisk support från Paris.

SCANTELE är idag generalagent för Epson-printer och LED i Skandinavien...

Med vänlig hälsning
SCANTELE AB

Karl-Axel Pettersson

Autostart på ABC 80

För de applikationer där det är önskvärt med att ett användardefinierat program startar direkt vid spänningstillslag eller reset finns nu speciella PROM att tillgå från både Scandia Metric och Sattco. Det nya PROMet ersätter det vanliga diskoperativsystemet (DOS) i FD2:an, FD4:an eller DataDisc80. Med det nya DOSet börjar ABC 80 att direkt vid tillslag leta efter ett program som kallas 'AUTSTART.SYS'. Om detta hittas kommer därefter sökning att ske efter 'BASICERR.SYS'. När så slutligen allt detta är slutfört sker den egentliga autostarten genom att starta användarens BASIC-program döpt till 'START'. Om inte programfilen 'AUTSTART.SYS' hittas vid uppstart kommer flexskiveenheten att fungera som vanligt, d v s måste anropas med önskat programnamn.

Själva modifieringen till automatisk start på en FD2a var klart enkel erfor ABC-KLUBBENS testgrupp då en ombyggnad genomfördes här för leden. Bara en enda skruv på flexskiveenhetens baksida behövde lossas för att enhetens kontrollkort skulle bli åtkomligt. När detta med ett enkelt handgrepp avskilt från sina två kontaktdon kunde det gamla PROMet enkelt ersättas med det nya. En manöver som på sin höjd tog ett par minuter. Inga lödningar eller kabelanslutningar o dyl behövdes således.

Så snart bytet av PROM-kapsel var genomfört var det bara att sätta tillbaka kontrollkortet på sin plats och ansluta dess kabelkontakt.

Vid köp av Autostartsatsen medföljer förutom ett PROM för FD2 eller FD4 (alt två PROMar för DataDiscen) även en programskiva innehållande programsekvensen 'AUTSTART.SYS'. Detta program ska sedan kopieras över till alla de skivor vilka ska utrustas med autostartoption.

Sådana här programtilläg kan i princip göras på vilka program som helst, men ett varningens ord bör dock ges för att försöka införa autostart på program av den typ som direkt gör anrop till det gamla DOSet utan att hoppa till hopptabel. Om uturen är framme kan då programmet bli okörbart.

BREV FRÅN SKOGEN

Av Gunnar Englund

Detta skulle kunna handla om problem med T80PRT och det gör det också. Dessutom handlar det om dokumentation, etik, nattsvart förtvivlan och mynnar slutligen ut i en kort presentation av ett skamligt okänt men oförsäkrat bra system för assembler-programmering.

Alltså, T80PRT är inte helt ofarligt. Som CRT-terminal är det OK men när det handlar om att ta in lite större datamängder utifrån så får man problem. T80PRT-manualen beskriver hur man tar in en sträng. Man kan bygga upp en lång sträng av delsträngar som kommer utifrån eller man kan dimensionera en strängvektor så att längsta förväntade sträng får plats. Mina strängar har högst varierande längd. Från några bytes (radnummer) till ca 60 bytes (fulla kommentarader).

För att undvika att omedelbart fylla upp minnet så är det nödvändigt att använda den förra varianten. Det är nu man får problem, stränghanteringen tar längre och längre tid ju längre strängen blir. Det är så, leverantören beklagar och beklagar.

Man kan komma en bit på väg om man kör med 110 Bd men så värst mycket mer än 5000 bytes får man inte ta in om det ska gå bra. Men 110 Baud - då kan man ju lika gärna köra teletype och stans.

Nu trodde jag faktiskt att det gick bra att köra i 110 Bd. Visserligen tog det tid men det heter ju "sakta men säkert" så egentligen var jag inte så bekymrad. Dumpning gör man ju i allmänhet bara när jobbet är klart och det finns så mycket annat att pyssla med medan data kryper fram på tampan. Så jag ordnade ett litet läckert bekvämlighetsprogram som hjälpte till att hålla reda på enheterna och blockera de värsta felkommandona och sen satte jag igång med det jag egentligen skulle göra dvs producera kod.

När timman blev sen så dumpade jag koden för att spara på skiva. Dumpningen såg ut att gå bra så jag gick ifrån ett tag och när jag kom tillbaka så hade jag nynn på skärmen och allt såg så bra ut. Ganska nöjd med dagen minns jag att jag var när jag bröt rubbet.

Den andra dagens morgon var det så dags att sätta fart på speleverket igen. Laddning av gårdagens data gick inte alls, koden blev illa. Tog fram data på skärmen så såg inte alls så tokiga ut. Rad efter rad kom upp och alla var de lika fina. Det var först efter ca 5000 tecken som det började se illa ut: rumphugget, stympat och förvrängt är lämpliga ord i sammanhanget.

T80PRT hade alltså börjat få problem att hinna med när strängen blev lite lång. Lägre hastigheter än 110 Bd finns inte i T80PRT och tur är det för annars skulle man väl få börja dumpningen vid lunch om man vill bli klar i anständig tid.

Jag såg ingen annan lösning än att göra en egen V24-mottagningsrutin som jag kunde anropa från Basicen vid behov. Den blev inte så lång så den fick plats i pokearean. T80PRT behöll jag för terminaldelens skull, jag uppskattar möjligheten att köra 80 tecken/rad. För övrigt ordnade jag så att jag i terminalmod kunde ge den yttre enheten besked om vad den skulle göra men istället för att avsluta med CR så lämnar jag terminalmod med ctrl P vilket gör att Basicen tar över och anropar hjälpsnutten. Denna i sin tur verkställer vad som förberetts i terminalmod genom att skicka ut CR och sedan snabbt som attan vända linjen och ta in data.

När data kommer så läggs de direkt upp i RAM, när överföringen är klar så kan man komma upp till Basic igen med ctrl A. Basic kan sedan läsa mottagna data och man är fri att i lugn och ro göra vad man önskar med dem. Rampekaren övervakas hela tiden så att risk för kollision med Basicvariablerna undviks. Hastigheten skulle kunna väljas högre än 2400 Bd men jag valde denna hastighet eftersom det är vad T80PRT klarar som mest.

Så långt funkade det hela (i teorin). I praktiken uppstod dock vissa problem, vid prov funkade inte T80PRT riktigt som det var tänkt. Styrfunktionerna för skärmen hade lagt av och vad värre var: ctrl P som ger återgång till Basic funkade inte heller.

Där sitter man alltså med ett helt låst system, det där jobbet som skulle ha varit klart i förrgår fortfarande ogjort och ingen ljusning vid horisonten. Nåja, reset funkade i alla fall...

Så det blev till att ladda igen och kolla vad som egentligen hände. Ganska snart (för nu hade adrenalinnet runnit till rejält) visade det sig att T80PRT har behov av RAM och det är ju i och för sig inte så häpnadsväckande. Det är bara det att T80PRT använder pokearean att lagra parametrar i. Det är nu man börjar fundera kring det här med etik. Är det VERKLIGEN NÖDVÄNDIGT att göra så? OK att det är bra att ha en fast plats när T80PRT 80 laddas till olika adresser, men Z80 har faktiskt en del möjligheter att klara även detta problem. Om det nu trots allt är nödvändigt att använda pokearean så borde det för bövelen stå en rad eller två om detta i manualen.

Plötsligt slog det slint totalt. I stället för att säga så här till mig själv: "Nu du Skogs-Gurra så får du assemblera om med ORG 32768 och lyfta BOFA ett hack" så sa jag så här: "Men - där ska jag ju ha ir T80PRT". Det är möjligt att det var stress, stora siffror på klockan och allmän irritation som spelade in.

Nu kommer det fina, nyttan av en användarklubb: jag ringde Tad G. som trots att det var ganska så sent försäkrade att Magnus Lundberg (se sid. 24 i nr 3 av klubbtidningen) gärna skulle prata lite kring detta. Denne förträfflige man vars psyke tydligen inte berörs av något så bara: "lyft upp BOFA och lägg din snutt längst ner". Det är detta jag menar med värdet av informella kontakter.

Jag gjorde så och se - nu kunde jag glädjas åt ett mycket vackert samarbete mellan enheterna.

"Soft and clean as a whistle" som utlänningarna skulle ha sagt.

ASMEDIT

När jag nu ändå har ordet så ska jag passa på att berätta om ett mycket trevligt men tyvärr rätt okänt system för assembler-programmering. Anledningen till att jag tar upp detta är att det var just till ett sådant system jag använde ABC80 och T80PRT i skildringen ovan. Det finns en anledning till, systemet har just blivit tillgängligt på skiva eller kassett för körning på ABC80.

Systemet heter ASMEDIT och finns dels som kretskort med CPU, RAM, I/O och programvara i EPROM för anslutning till INTELs utvecklingshjälpmedel "PROMPT" för 8080,8048 etc. dels som program att laddas i ABC 80.

Det är SYNTHESSE i Lund som har tagit fram systemet och det säljs av Nordisk Elektronik. Redan namnet säger att det handlar om en assembler och en editor men det är inte vilken assembler som helst och knappast heller vilken editor som helst. Editorn har bland annat den egenheten att den inte accepterar någon annan text än absolut rätt (bortsett från antal mellanslag) skrivna mnemoniska instruktioner plus ett antal pseudoinstruktioner.

Knasigt? Nej, mycket bra. Man kan inte göra fel. När editorn märker att en instruktion är fel inknappad så ber den vänligt att få in den en gång till, om den är rätt denna gång så läggs den till källkoden och nästa radnummer kommer upp.

Hur kan editorn veta att en instruktion är rätt eller fel? Jo, editorn gör större delen av assemblerns jobb genom att göra så mycket av översättningen som den kan. Om den inte klarar av det den vet att den bör klara så kan den bara dra slutsatsen: fel i källkoden. Det verkar bekant, inte sant? Jomennisst, det är ju precis så ABC 80 arbetar.

Vad innebär nu detta utöver att man förhindras att skriva fel? Jo, assembleringen går mycket snabbt. Allt assemblern behöver göra är ju att leta reda på lablarnas adresser och symbolernas värden. Men vart har tiden tagit vägen då? Frågar vetgirig. Man brukar ju sällan få något gratis.

SYNTHESSE uttrycker det så här: "While you scratch your head ASMEDIT works" vilket alltså innebär att systemet utnyttjar de millisekunder som operatören tänker innan han knappar in nästa rad.

Assemblern gör lite mer än så. Vissa feltyper kan konstateras först vid assemblering, dit hör bland annat saknad eller dubbelt specad label/symbol, för långa hopp och så vidare. När man får ett felmeddelande från assemblern så måste man rätta källkoden. Detta brukar innebära att editorn ska laddas, källkoden hämtas in och rättas, sparas på skiva, assemblern in, assemblera etc. etc. En operation som i många system kan ta uppemot en halvtimme. Så icke i ASMEDIT. Editor och assembler är residenta och källkoden ligger ju hela tiden i RAM snabbt åtkomlig.

För att göra det hela ännu lite snabbare så används så gott som genomgående "enknappskommandon".

Det är egentligen bara kommandot "NEW" som man inte kan nå av misstag och det tycker jag är riktigt bra. Rättning av källkod och assemblering går alltså mycket snabbt. Att skriva in en ny rad med källkod tar bara några sekunder och assembleringen går lika snabbt. Jo, det är sant. Det tar faktiskt inte mer än 4 till 5 sekunder att assemblera ett lk program. Det är visserligen sant att lk objektkod inte är särskilt mycket men som modul betraktat är det inte illa. Måkan faktiskt tala om interaktiv assembler-programmering.

LÄGET PÅ PROGRAMFRONTEN

Det verkligt trevliga börjar när koden ska testas mot hårdvaran. I PROMPT kan programmet köras mot aktuell applikation genom att man kopplar en navelsträng till CPU-sockeln på aktuellt system och kör. PROMPT har en del debugmöjligheter men kan inte köras i realtid med brytpunkter. Genom att tillfälligt lägga in stoppsnurror och/eller spara data i intressanta delar av programmet så klarar man även tidskritiska avsnitt.

I ett vanligt system drar man sig i det längsta för att föra in sådana tillfälliga ändringar i ett program som är assemblerat, antingen ska man föra in tilläggen för hand vilket är både jobbigt och riskabelt eller också ska man ändra källkoden och assemblera om, det tar tid som sagt och man gör nog inte så många ändringar innan man börjar tycka att man fått nog.

I ASMEDIT lägger man in vad man vill, helst man vill. Det tar längre tid att redigera ut vad man vill lägga in än att göra det och assemblera om. Det gör avlusningen rolig och effektiv, man behöver inte tänka att "det vore intressant att se vad som händer om..." man tar helt enkelt reda på "vad som händer om...".

Det allra trevligaste kommer sist: när programmet löper som det ska så är också källkoden korrekt. Inga glömda patchar och dokumentationen är OK.

ASMEDIT är alltså mycket lämpligt för mindre tillämpningar, upp till 3-4 k speciellt om man jobbar med moduler. För större program bör man kanske tänka sig ett mera normalt system. Det blir lite jobbigt att hålla reda på alla yttre lablar annars.

Ovan sagda gäller system på kort, jag har inte haft tillfälle att prova den "mjuka" varianten men jag föreställer mig att den är minst lika trevlig och förmodligen trevligare eftersom den är något år yngre.

Jag hoppas kunna återkomma med rapport framöver. ■

Rapport från amatörradiosektionens möte den 3 november

Ändagen den 3/11 hade vi vårt andra möte, men nu i de nya lokalerna i Bromma. Ett 35-tal medlemmar ställde upp och det blev en demonstration av RTTY med två kompletta anläggningar bestående av ABC 80 med RTTY-modem samt FM-transciever för 145 MHz. Kommunikation fördes inom lokalen helt trådlöst via radio.

Senare kom även SM5DWC Anders P., medlem i Huddinge, igång och ett QSO (samtal dvs tittande) kördes och det hela fungerade till allas belåtenhet. Frekvensen som utnyttjades var 145,300 MHz frekvensmodulerat och är den standardfrekvens som är till för RTTY. Modemen som användes är ihopsnickrade av SMOHPP Mats, SMOGBU Berndt, SMOLKE Owe samt SM5DWC Anders, vilka också genomförde demonstrationen. Modemen är specialbyggda för RTTY men utnyttjar telefonmodemkretsar. Tonparen som användes var 1275 resp 1445 Hz med hastigheten 45 baud Baudot-kod. Program och ritningar på modemet kommer att publiceras i ABC-bladet inom kort.

Väl mött och 73 samt en trevlig RY-jul önskar M.G.
Amatörradiosektionen.

Diskussion om programvaran har funnits lika länge som datorerna själva. Och fortfarande är problematiken lika aktuell. För vår del gäller det ABC 80. Nedan följer delar av ett cirkulärbrev från Datakraft i Lund. Vi publicerar det för att få igång en konstruktiv och givande debatt. Alla är välkomna med kommentarer och förslag.

"ADMINISTRATIVA PROGRAM FÖR ABC80"

ABC 80 har nu sålts i bortåt 12 000 exemplar, till stor del för olika administrativa ändamål. Behovet av sådana program är alltså mycket betydande. Tyvärr är läget på denna front ganska genant, det är ont om program av god kvalitet. Problemet är inte unikt för ABC 80, för många andra mikrodatörer är situationen rent katastrofal. Bland bristerna brukar märkas att programmen inte är ordentligt systemerade, svåra att underhålla, saknar dokumentation av tillbörlig kvalitet och har låg driftsäkerhet, samt sällan uppfyller vare sig god revisionsred eller god datased.

Läget är sålunda sådant att ABC 80 långsamt men säkert håller på att få dåligt rykte som administrativ maskin.

DATAKRAFT AB har därför beslutat att ta fram en programserie benämnd MODUL 81. Siktet är dessutom inställt på en helhetslösning som innefattar handböcker, utbildning, stöd till lokala distributörer mm.

Projektet har övervägts länge inom DATAKRAFT. Det är nämligen i hög grad resurs- och tidskrävande. En bra leverantör måste behärska såväl systemering som programmering, ha tillgång till ekonomisk och administrativ expertis, samt ha bred erfarenhet av såväl dokumentation som utbildning. Vi anser att DATAKRAFT AB, som har levererat över 200 datasystem av skiftande storlek och komplexitet, är lämpad för uppgiften...

Följande punkter kommer bli att tas upp av DATAKRAFT i projektet MODUL 81:

Genomgång av marknadens krav. Utformning av huvudspecifikation. Systemering av ett programpaket enligt IDK-metod.

Programmering av de ingående programmen. Högst sannolikt kommer programmeringen att läsas till CPT (Chief Programming Teams) och JSP.

Skydd av programvara har ägnats särskild uppmärksamhet. Flertalet program kommer att vara listbara och ändringsbara. Däremot kommer alla program att vara kopieringsskyddade. Hänsyn till maskinvaran har givetvis ägnats stor uppmärksamhet. Programsystemet är körbart på ABC 80 i standardutförande utan minnesexpansion, i stort sett valfri skrivare och i n på följande flexskiveenheter: FD2, FD2U, FD4, FD4D, MP2, MP2D. Programmen är fullt körbara under såväl nya som gamla DOS.

Resursåtgången för MODUL 81 beräknas till 2-2.5 månår.

Det kan tilläggas, att av den kommande serien är Lager 81 redan tillgänglig och redovisning kommer inom kort.

Bland insändare i nr 3 av ABC-bladet fanns ett brev, som vi fick av Kaj Arnö från Finland. Då insändaren den gången handlade om användarnas syn på ABC 80 har vissa delar, inte relaterade till det, utlämnats. Här kommer en sådan del som vi finner intressant och därför publicerar. Som framgår av det efterföljande programmet har Kajs satser av Bengt Olwig gjorts till ett litet program.

"..Själva ABC-bladet verkar vara en mycket användbar och upplysande tidning för oss ABC-ägare och ABC-klubbsmedlemmar. Redan på sidan 2 (ABC-bladet nr 1 - red ann) fick vi oss en hel del nyttig information, programmeringstipsen. Denna spalt kunde gärna utökas betydligt, och jag bidrar gärna med en funktion:

```
10 DEFFNK=((PEEK(-526%) XOR 255%)*256%+(
PEEK(-527%) XOR 255%))*5.12+(PEEK(-52
8%) XOR 255%)/50
```

Denna funktion för klockan avläses enkelt och flärdfritt med ;FNK, t ex. Vill man nollställa klockan någon gång går det bra med POKE 65008,-1,-1,-1.

EXTRACT och VARIABEL är två ytterst användbara program, som väl varje ABC-ägare tackar för. Men för att flera sådana program skall kunna göras borde ytterligare information om ABC:s systemvariabler publiceras, i likhet med det vi fick veta på sid 14.

```
10 REM *****
20 REM * TIDEN.BAC *
30 REM * VER 1.0 / 1980-12-02 *
40 REM * (C) ----- *
50 REM * gjort av Kaj Arnö *
60 REM * mod av Bengt Olwig *
70 REM * insänt av ----- *
80 REM *****
90 REM
100 REM * DEFINIERING AV TIDSALGORITMEN
*
110 DEFFNK=((PEEK(-526%) XOR 255%)*256%+
(PEEK(-527%) XOR 255%))*5.12+(PEEK(-
528%) XOR 255%)/50
120 REM
130 REM * HÄR NOLLSTÄLLS KLOCKAN OCH BIL
DSKÄRMEN RENSAS *
140 POKE 65008,-1,-1,-1 ; CHR$(12%)
150 REM
160 REM * HÄR SKRIVS TIDEN UT *
170 ; CUR(10%,19%);FNK
180 GOTO 160
```

Tävling

16 000 kronor tre grabbar Tävligen, som samarbetet med götlands län, från hela lan

"Vi visste, ungdomar, och så dukta säger Martin Tävligen, intresset för sig till ungdom var att både skulle få upp Därför var för att verk skolan och datorverksam och Data Sae programmerar Ekhagen från lands län och och Luxor's p

Priserna d i Stockholm : Utvecklingsf talade om c samband med d

Första p kronor, mott Karl Kahlau, Hooper. Pri den klass c går. Det är hur pengarna

Programm presenteras s "Tanken t med ABC 80 nytt progra kommer här JSP-Basic. Pr mera struktur överskådliga. språk om man JSP-metoden, det så bör r och skriva pr

För de mer att programr läser en fil omvandlar den filen) med va lerat format.

Hur komrn gör ett sådan

"Vi läste o Det fanns ju Basic på AB behövdes fä Siktade gjord bästa. "Mot och skolan ha Adam, som 80 med en s intresserad som köpte s har sysslat r Michael, 16, har hunnit r på Computerk Alla tre är r av Unga Fors

Tävlingar för bästa ABC-80-programmet avgjort

16 000 kronor i prispengar har gått till tre grabbar från Hersbyskolan på Lidingö. Tävligen, som utlystes i våras av Luxor i samarbetet med Utvecklingsfonden i Östergötlands län, resulterade i 300 insända bidrag från hela landet.

"Vi visste, att det fanns många duktiga ungdomar, men att det fanns så många och så duktiga - det hade vi inte anat", säger Martin Graap från Luxor.

Tävlingen, som utlystes för att stimulera intresset för datorer i undervisningen, vände sig till ungdom på gymnasienivå, men tanken var att både lärare, elever och de ansvariga skulle få upp ögonen på "datorrevolutionen". Därför var även Utvecklingsfonden med - för att verka som en slags länk mellan skolan och näringslivet och stimulera länets datorverksamhet (där finns ju både Luxor och Data Saab med stora behov av duktiga programmerare). I juryn fanns, förutom Sture Ekhamen från Utvecklingsfonden i Östergötlands län och Martin Graap, även Åke Westh och Luxor's programvaruchef Alf Petersson.

Priserna delades ut på Tekniska Mässan i Stockholm av ordföranden i Östergötlands Utvecklingsfond Hilding Ferm, som bl a talade om det stora utbildningsbehovet i samband med den snabba tekniska utvecklingen.

Första priset, en check på 16.000 kronor, mottogs av det vinnande laget - Karl Kahlau, Adam Horvath och Michael Hooper. Priset tillfaller alla eleverna i den klass där Karl, Adam och Michael går. Det är också klassen som bestämmer hur pengarna ska användas.

Programmet som vann heter JSP och presenteras så här av författarna:

"Tanken bakom detta program var att med ABC 80's Basic som grund skapa delvis nytt programspråk. Detta programspråk kommer här i fortsättningen att kallas för JSP-Basic. Program skrivna i JSP-Basic blir mera strukturerade, användarvänliga och lättöverskådliga. Särskilt kraftfullt blir detta språk om man kan strukturera program enligt JSP-metoden, men även om man inte kan det så bör man med fördel kunna utveckla och skriva program i JSP-Basic."

För de mera dataintresserade kan tillägas, att programmet är en sk preprocessor. Det läser en fil (källfilen) med JSP-Basic och omvandlar den till en motsvarande fil (objektfilen) med vanlig ABC 80's Basic i komplicerat format. Detta om det tekniska.

Hur kommer det sig att tre unga killar gör ett sådant program?

"Vi läste om JSP, tyckte det var vettigt. Det fanns ju för Cobol, så varför inte för Basic på ABC 80?" Tävligen var vad som behövdes för att de skulle sätta igång. Siktade gjorde de på att vara bland de tio bästa. "Mot slutet blev det ganska jobbigt, och skolan har väl lidit något också", säger Adam, som är 16 år, har en lånad ABC 80 med en skivenhet och har blivit datorintresserad för två år sedan. Karl, 17, som köpte sin ABC 80 för egna pengar, har sysslat med data i mindre än ett år. Michael, 16, har ingen egen maskin men har hunnit med att i åtta månader arbeta på Computerland, en datorbutik i Stockholm. Alla tre är med i Lidingös lokal-avdelning av Unga Forskare.

Andrapriset på 6 000 kronor gick till Bjarne Borg och Ted Lindström från Fridsgårdsskolan i Bålsta utanför Stockholm. Bjarne, som är 15 år och har en egen ABC 80, har varit dataintresserad i ett år. Ted, 14, har ingen egen maskin utan arbetar tillsammans med Bjarne. De tyckte att det var roligt att tävligen utlystes och för att ligga bra till skickade de in 10 olika program. Under tiden som gått har de hunnit med en hel del spelprogram, registerprogram och ett enkelt bokföringsprogram.

"Det vinnande registerprogrammet gjordes under sommaren och hösten och vi höll på att missa sista inlämningsdagen" - säger Bjarne.

Hedersomnämmandet för bästa spelprogram har gått till Anders Andersson, Lars Oestreicher och Tomas Svensson från Sandbroskolan i Enköping för programmet Hospital. Ett bra initiativ, som visar att det där med datorer inte är svårt ens för mycket unga människor. Ett initiativ som bör följas av andra för att få bort den rädsla och mistro som finns mot den nya tekniken i dagens samhälle.

RESULTAT I ABC 80 - TÄVLINGEN

KLASSVINNARE (1 000:-)

1. Administrativ klass/högstadiet

"REGISTER", Bjarne Borg, Ted Lindström
8 D, Fridsgårdsskolan, Bålsta

2. Teknisk-matematisk klass/högstadiet

"ANDRAGRADSEKVATIONER", Mats Ulleryd, Kristina Ström, 9 F, Gullmarsskolan, Lysekil

3. Öppen klass/högstadiet

"LASER", Måns Henningson, Andreas Bladh, Åk 9, Norrreportskolan, Ystad

4. Administrativ klass/gymnasiet

"SIMNING", Mats Nygren, Peter Ottosson, Åk 3 teknisk linje, Soltorgsskolan, Borlänge

5. Teknisk-matematisk klass/gymnasiet

"PP", Stefan Pettersson, Jan Johansson, Pontus de Verdier, NT 1, Vasagymnasiet, Arboga

Öppen klass/gymnasiet

"JSP", Adam Horvath, Karl Kahlau, Michael Hooper, T1E, Hersby skola, Lidingö

Hedersomnämmande:

"Bästa spelprogram": HOSPITAL; Anders Andersson, Lars Oestreicher, Tomas Svensson, NA3, Sandbroskolan, Enköping

B. Gruppvinnare (5 000:-)

"REGISTER"

Bjarne Borg, Ted Lindström, 8 D, Fridsgårdsskolan, Bålsta

"JSP"

Adam Horvath, Karl Kahlau, Michael Hooper, T1E, Hersby skola, Lidingö

C. Totalsegrare (10 000:-)

"JSP" (tot 16 000:-)

Adam Horvath, Karl Kahlau, Michael Hooper, T1E, Hersby skola, Lidingö



Hilding Ferm delar ut första priset till Michael Hooper, Adam Horvath och Karl Kahlau

Verksamhetsberättelse för ABC-klubben 1980

ABC-klubben bildades genom val av en interimstyrelse vid ett möte på KTH i Stockholm den 24 januari 1980. Till interimstyrelsens ordförande valdes Odd Rolander. Det bildades också ett antal intressegrupper samt utsågs ledare till dessa. Vid det första årsmötet den 22 april godkändes föreningens stadgar och budget. Årsavgiften fastställdes för 75 kr för seniorer och 40 kr för juniorer. Till styrelse i ABC-klubben valdes:

Gunnar Tidner	ordförande
Göran Österman	vice ordf
Bengt Olwig	kassör
Hans Nilsson	sekreterare
Tad Gruber	redaktör
Jockum Wahlberg	
Veikko Honkamäki	suppleant
John Kvarnström	suppleant

Styrelsen har under verksamhetsåret haft 14 protokollförda sammanträden. John Kvarnström har på egen begäran lämnat styrelsen under hösten. Till ny suppleant efter honom har Ulf Sjöstrand adjungerats. Som ledare för intressegrupper har under året fungerat:

Småföretagare/ADB - Odd Rolander, Programmeringsteknik - Göran Österman, Datakommunikation - Hans Nilsson, Ordbehandling - Olof Söderlind, Styr- och mät/hårdvara - Bengt Olwig, Medicin - Owe Schill, Amatörradio - Mats Garle, Musik och grafik - Göran Sundqvist, Schack - Jockum Wahlberg.

Vid verksamhetsårets utgång hade klubben 1432 medlemmar. En matrikel över klubbens medlemmar, sorterade i alfabetisk ordning och postnummerordning har utgivits.

Klubbens främsta ansikte utåt har varit tidningen ABC-bladet. Redaktören Tad Gruber har under året haft hjälp av en redaktions-

kommitte bestående av Bengt Olwig, Odd Rolander och Göran Österman. Tidningen har utkommit med 4 nummer, tryckta i en genomsnittlig upplaga på 3500 exemplar.

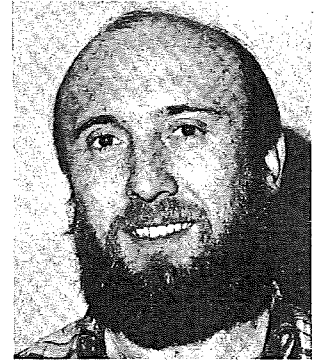
Efter lång tids sökande fick klubben äntligen tag på en klubblokal i medborgarhuset vid Alvik i Bromma. Lokalen består av tre mindre rum på sammanlagt 43 kvm. I huset finns tillgång till större samlingslokaler för klubbmöten mm. Till klubblokalen har anskaffats två st telefonabonnemang, 08/80 15 22 och 80 15 23. På det ena finns en telefonsvarare med aktuell klubbinformation. På den andra finns ett modem med automat-svar och en ABC-80 inkopplad för överföring av program till klubbens medlemmar. Utrustning i klubblokalen har ställts till klubbens förfogande av bl a Luxor AB och SATTCO.

Under hösten har anordnats ett antal klubbträffar i samlingslokalerna i medborgarhuset vid Alvik. Förutom två allmänna möten i Brommasalen den 10 okt resp 14 nov, det senare på temat grafik och musik, har det varit temakvällar ett antal måndagar i den mindre Vidängssalen. Som tema har förekommit modemkommunikation, amatörradio, programbyarkväll, textbehandling, diskussion om upplägnig av studiecirkel i programmering, musik och hårdvaran i ABC-80. Klubbens styrelse har tagit initiativ till en försöksverksamhet med kassettkopiering för distribution av programvara till klubbens medlemmar.

Stockholm 1981-02-01

Gunnar Tidner
Bengt Olwig
Tad Gruber
Veikko Honkamäki

Göran Österman
Hans Nilsson
Jockum Wahlberg
Ulf Sjöstrand



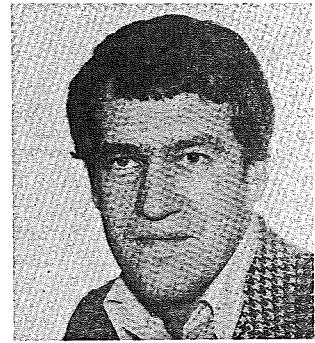
Tad Gruber



Göran Österman



Hans Nilsson



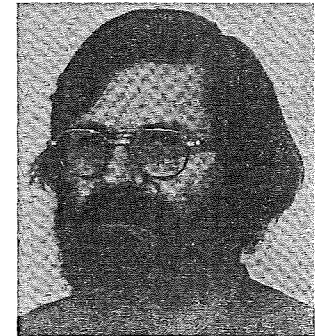
Jockum Wahlberg



Gunnar Tidner



Bengt Olwig



Veikko Honkamäki