

ABC BLADET

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

Nummer 2, 1982



DATAINDUSTRIER AB är ett utvecklingsföretag som under de senaste åren lanserat en rad uppmärksammade produkter varav ABC 80 och ABC 800, som tillverkas av LUXOR, och DATABOARD 4680 är de mest kända. Detta samt mycket annat har vi åstadkommit med 5 duktiga konstruktörer och 15 skärpta programmerare.

Vi söker

SOFTWARESPECIALISTER

som vill vara med där både det mesta
och bästa händer inom landets

MIKRODATORUTVECKLING

Förutom vidareutveckling av ABC 80/800 och DATABOARD har vi just påbörjat flera projekt utomlands. Vi måste förstärka framförallt vår programmeringssida med ytterligare softwarespecialister.

Om du har brinnande intresse och dokumenterad läggning för programmering i assembler får du chansen att syssla med kompilatorer, operativsystem eller intressanta applikationer med distribuerade datorer. Vi arbetar även i BASIC, PASCAL och C.

Du kommer att arbeta och vidareutvecklas i ett gäng av kreativa lite mikrodatortokiga människor. Vi håller till i en ny fastighet i Täby där vi har massor av datorer och andra hjälpmedel.

Vi har även möjlighet att ordna examensarbete för dig som ännu ej avlagt examen.

Hör av dig för ytterligare information och eventuellt besök till Harrgot Lindmark eller Ulf Nordberg på tel. 08-768 06 60 eller skicka in dina handlingar till

**DATA
INDUSTRIER AB**
Enhagsvägen 9, 183 30 TÄBY

Medlemsorgan för ABC-klubben

Vidängsvägen 1
161 33 Bromma
ISSN 0349-3653

Ansvarig utgivare:
Gunnar Tidner

Redaktör:
Claes Schibler

Monteringshjälp Stig Löfgren

ABC-klubbens
postgiro 15 33 36-3
telefon 08-80 15 22
(automatisk telefonsvarare
med aktuell klubbinformation)
telefon 08-80 15 23
(modem med ABC Monitor)

Tryck: Märstatryck AB 1982

Nummer 2, 1982

INNEHÅLL

Omslagsbild CASIO Göran Sundqvist	
Ledare av Gunnar Tidner	2
En englandsresa av Göran Sundqvist	3
Mera om TV-editorn av Gunnar Tidner	6
Forth för ABC80 av Torbjörn Alm	8
Här är Fig-FORTH av Gunnar Tidner	9
ABC80 blir RAM-maskin av Kjell-Åke Johansson	10
Kassett ett + två + tre	12-14
Svenne av Kjell-Åke Johansson	13
Brev om fel i ABC80 Bernhard Granholm	16
Q-Zentralen	17
Möten i Stockholmskretsen	17
Kontaktproblem Lars Frej	18
Screena program	19
Pausbird av Ferdinand Mican	20
Köranvisning ABCV24 Scandia Metric	22
Diskett och Programmärkning	24
Sätt autopilot på din ABC80 Sven Wickberg	28
Kortlek + Personnummer Sven Wickberg	30
Stortext + Floppyister Sven Wickberg	32

Annonsspriser fr o m 1 juli 1980

1/1-sida 185 × 260 mm	2000:-
1/2-sida 185 × 128 eller 90 × 260 mm	1.200:-
1/3-sida 185 × 85 mm	800:-
1/4-sida 90 × 128 mm	650:-
2 st 1/1-sidor i uppslag	4.500:-
2:a omslagssida	2.500:-
3:e omslagssida	2.300:-

4:e omslagssida 185 × 225 mm 2.750:-
Begärd placering 10% förhöjning.

Tidningen ansvarar ej för att införda programlistningar är korrekta.

Särskild prislista vid best. av flera ex. tillhandahålles på begäran.

Copyright gäller för införda program om inget annat anges.

Medlemsavgifter 1982

Seniorer 125 Skr
Juniorer 70 Skr

Junior räknas man t o m det kalenderår man fyller 18 år. Ange därför personnummer när Du betalar medlemsavgiften. Medlemskapet är personligt. Särskild prislista för prenumeration tillhandahålles på begäran.

Medlem blir Du enklast genom att sätta in medlemsavgiften på ABC-klubbens postgirokonto 15 33 36-3.

ABC-klubbens styrelse 1982

Ordförande: Gunnar Tidner
Vice ordförande: Kjell-Åke Johansson
Sekreterare: Ulf Sjöstrand
Kassör: Marianne Forsman
Redaktör: Claes Schibler
Ledamot: Joe Johnsson
Suppleanter: Allan Larsson
Björn Sjöborg
Göran Sundqvist

LEDAREN

Ledaren

Det händer att man träffar folk som fnysr åt ABC-80 och undrar hur man kan stå ut med en "smådator" som är närmare fyra år gammal. Vad de inte tänker på eller inte vet om är att jämförelsen med andra system tillverkade inom eller utom landet ofta utfaller till ABC-80:s fördel när man tar hänsyn till alla faktorer inte minst ekonomiska.

När jag talar om ABC-80 så menar jag hela **ABC80-systemet**, alltså inte enbart grundenheten utan även all den kringutrustning som kan anslutas till systemet. Under de fyra åren har det skett en enorm teknisk utveckling med en kraftig prestandahöjning såväl av grundenhet som kringutrustning (printrar, flexskiveutrustning och andra massminnen, fleranvändarsystem, plottrar och mät- styr- och signalutrustningar mm).

För många är det numera standard att tangentbordet har inbyggt 32 K extra minne, 80-teckens bildskärm och ev andra tillsatser.

Vad som är väsentligt när det gäller den utveckling som ägt rum är att det kunnat ske med bibehållen **kompabilitet nedåt**. Med detta menas att program och annat som utvecklats på en enklare utrustning utan vidare går att använda när man övergår till utrustning med mer överlägsna prestanda.

Självklart skall man i ABC80-systemet räkna in vad som under åren gjorts på programvarusidan både av amatörer och mer professionella programmerare. Jag vill hävda att man i ABC80-systemet även bör räkna in **allt det kunnande som byggts upp hos användarna**.

ABC-klubben har sett det som en av sina främsta uppgifter att ta vara på och förmedla dessa erfarenheter, det tekniska kunnande och den insikt i programmering som finns bland klubbens medlemmar.

Vi försöker göra det på flera olika sätt: genom denna tidning ABC-bladet, genom den påbörjade serien av ABC-rapporter, genom att på kassett distribuera program som gjorts av andra medlemmar eller som utvecklats direkt på ABC-klubbens uppdrag eller som vi på annat sätt fått tillgång till, genom monitorn med program- och informationsbank, och nu senast Q-Zentralen som efter överenskommelse blivit tillgänglig för ABC-klubbens medlemmar.

Ett av de projekt som ABC-klubben stöttat är den Fig-FORTH som nu ställs till klubbmedlemmarnas fria disposition. FORTH distribueras på ABC-kassett nr 4 och 5. Därigenom får alla som är medlemmar ABC-klubben tillgång till det nya programspråket. Samtidigt har vi tagit fram en dokumentation som finns att tillgå i form av en ABC-rapport. På liknande sätt kommer vi förmodligen att göra med andra kvalitetsprogram där dokumentation blir alltför omfattande för att i sin helhet återges i ABC-bladet: Programvaran blir tillgänglig för alla genom att den läggs på kassett men dokumentationen ges ut separat i form av ABC-rapport.

I FORTH har ABC-klubben fått en **verklig godbit**, det är initierade bedömmare överens om. Det dröjer kanske innan du upptäcker vad det är du fått, men bevara programmet väl, det kommer kanske en dag då du kommer att känna den verkliga upptäckarglädjen. FORTH är ingen leksak. Det är **nyttoprodukt** som passar väl på ABC80 för olika tillämpningar. Vilka kommer säkert att framgå av kommande ABC-blad.

I motsats till FORTH får nog den **LISP** som kommer på ABC-kassett nr 6 sägas vara huvudsakligen en leksak. Men eftersom ABC-klubben enligt sina stadgar skall verka för bl a ökade datorkunskaper till nytta, utbildning och nöje så försvarar även LISP en plats på kassetten. Om du tycker att FORTH är ett konstigt språk så är LISP ännu konstigare. Men det hjälper dig kanske att inse att BASIC inte är allt när det gäller programspråk.

ABC-klubben har blivit en rikstäckande förening, den är utan tvekan landets största datorklubb. **Mötesverksamhet** på olika platser i landet är ett ofta uttalat önskemål. Det låter sig tyvärr inte göras att driva mötesverksamhet i riksföreningens regi. Detta måste ankomma på lokalavdelningar av ABC-klubben att anordna. Det händer tyvärr ingenting om inte några är beredda att lokalt ta initiativ till att samlas. Använd matrikeln för att se vilket underlag som finns på din ort. Sätt inte ambitionerna för högt i början! Det viktiga är att **skapa ett forum där intresserade kan träffas och prata om sina olika problem och få hjälp av varandra**. När mötesverksamheten väl kommit i gång kan den utvecklas och resulterar kanske i bildandet av en lokalavdelning. Det kanske bildas studiecirklar eller annan organiserad kursverksamhet. Tänk på att försöka tillgodose behoven även hos de som inte har så stora förkunskaper. Det väsentliga är att komma igång. Låt inte det bästa bli det godas fiende.

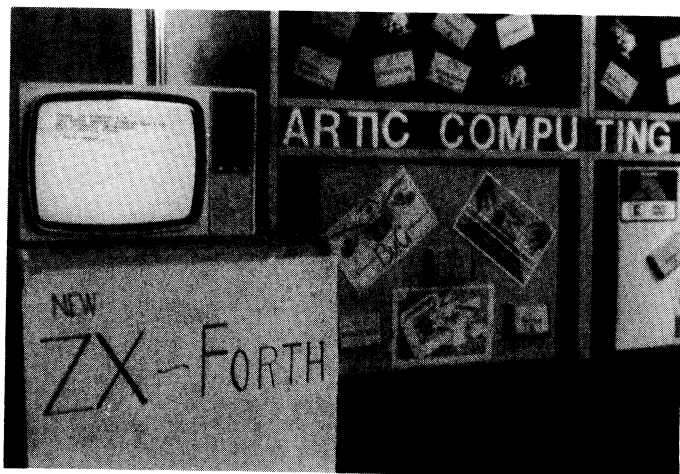
Vi som har hållit på länge och varit med om hela utvecklingen på ABC80-området har ett speciellt ansvar att hjälpa till med att förmedla kunskaperna till alla de som inte har så lång tids erfarenhet.

I kommande nummer av ABC-bladet kommer vi att ägna större utrymme åt den typ av problem och frågeställningar som möter nybörjaren.

Gunnar Tidner

En Englandsresa

Göran Sundqvist



ARTIC COMPUTING Ltd (tfn 00944-482 75284) hade implementerat FORTH på ZX 81. RAM-versionen kostar 35 pund. ROM-versionen som ersätter BASIC-ROMet kostar 69,95 pund. Forts. sid 4-5

VÄLJA FRITT ÄR STORT

Monroe
if 800
ABC 80/800
PET 4000/8000
Canon CX-1
VIC
CPM

Osborn
Sharp PC 1500
DataVision
UNI 80/800
TKN 80
Enkortsdatorer
Skrivare

Plotters
Program
Disketter
Promprogram
DataBoard
Litteratur
Papper m m



VÄLJA RÄTT ÄR STÖRRE

Slink in! Prova! Jämför!

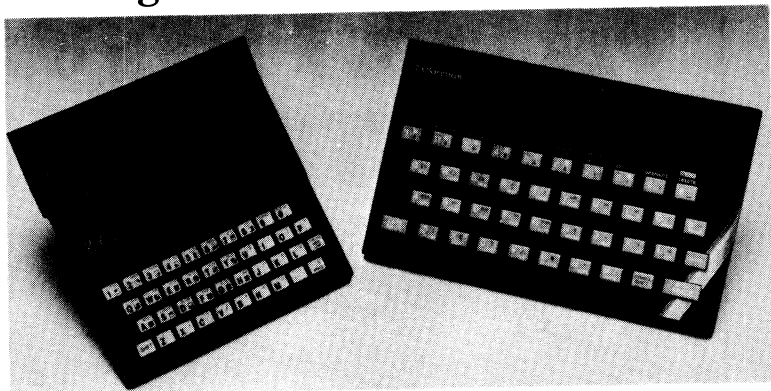
T-D-X

SmåDatorerAB

T-D-X SMÅDATORER AB KUNGSGATAN 79 · 112 27 STOCKHOLM
TEL 08-52 1060 · 52 1460

Skylltönstret för smådatorutvecklingen.

En Englandsresa (forts.)



Sinclair ZX Spectrum

Sinclair har flera smådatorer: den nya Sinclair ZX Spectrum (t h) och Sinclair ZX81 (t v) som har sålts i mer än 350 000 exemplar över hela världen.

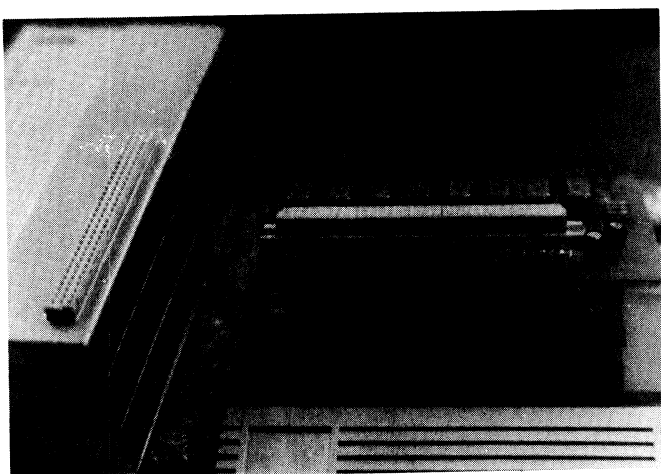
Måtten är 233 x 144 x 30 mm och den kan fås i två versioner, den ena med 16 k RAM för 125 pund och den andra med 48 k för 175 pund, detta inklusive den engelska momsen.

ZX81 finns i byggbits-sats för 69.95 pund och i byggsats för 49.95 pund, även detta inklusive den engelska momsen.

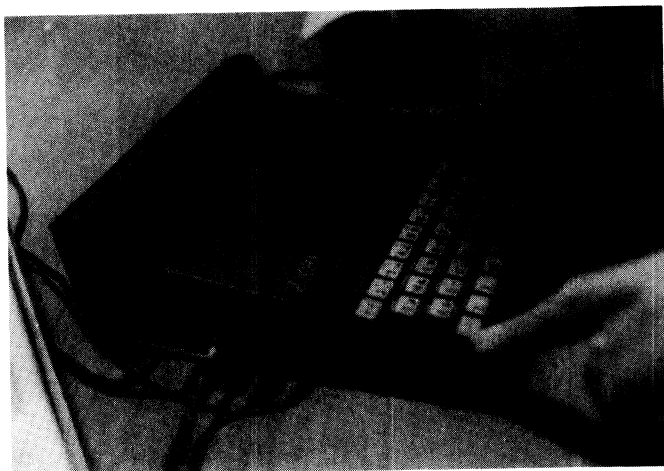
Upplysningar kan fås från **Daniel J Edelman Ltd**, Stanhope House, Stanhope Place, London W2 2HH eller direkt från **Sinclair Research Ltd**, 6 King's Parade, Cambridge CB2 1SN.



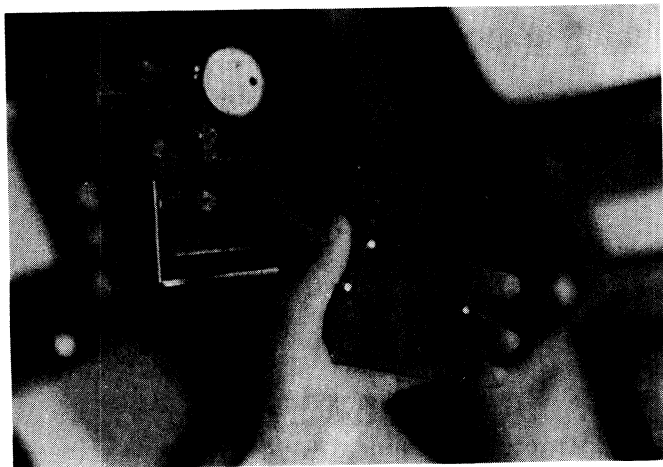
"The little professor", bygglåda för experiment med mikrodator.



Basicare Microsystem Ltd visar ett unikt patentsökt system för moduluppbyggnad av kretskort.



Memopak 64 k, 64 k RAM till ZX81 från **Memotech Ltd**, Oxford kostar 79.00 pund. Deras 16 k RAM kostar 39.95 pund. Snart kan man också erbjuda högupplösningsgrafik och anslutningselektronik för skrivare.



Den Ungerska "Godbiten" Mikrodiskettdrive och Mikrodiskett



En Englands- resa

En vecka i slutet av april fick jag tillfälle att tillbringa i England.

Under den veckan kunde jag pricka in inte mindre än tre elektronik- och datamässor plus Bachs h-mollmessa och mycket annat ur Londons enorma utbud. T.ex. alla elektronik- och datorbutiker i södra änden av Tottenham Court road med utgångspunkt från Oxford street. Närmaste tunnelbanestationen heter Tottenham Court road. En annan lämplig utflyktsgata är Edgware road, där man också kan hitta affärer som säljer komponenter och surplus. Även här börjar man promenaden från Oxford street, tunnelbanestation Marble Arch.

Första utställningen jag besökte var Communications 82 i Birmingham. Där hade man samlat ihop en ansevärd mängd fackfolk. På utställningen fanns praktiskt taget allt representerat som kan inrymmas under begreppet kommunikation.

Bland de större ämnesgrupperna kan nämnas:

-Antenner och antenntillbehör.

Här hittade jag det enda svenska företaget Allgon Antenn AB från Åkersberga. Övriga svenskar fanns representerade av distributörer och dotterbolag. T.ex. Thorn Ericsson och Ericsson Information Systems Ltd.

-Radiokommunikation.

Storbritannien har nyligen fått ett privatradioband öppnat för allmänheten och sådan utrustning var naturligtvis av stort intresse. Professionell kommunikationsradio var annars en av huvudgrupperna på utställningen.

-Telex

Telex var fortfarande viktigaste mediet för överföring av skriven text, även om teletex och telefax visades hos några utställare.

-Telefoner

En annan huvudgrupp var naturligtvis telefoner och växlar. Man får hoppas att Svenska Televerket hade folk över som fick se hur en modern myntapparat skall se ut. De som visades på mässan hade alla det gemensamt att man på en sifvertabell presenterade hur tillgängligt penningbelopp minskade allt efter samtals-tiden löpte.

-Datakommunikation

Huvudvikten låg här på MODEM och testutrustningar. Flera företag visade handburna, batteridrivna utrustningar för test av datalinjer.

Trend visade ett nytt koncept, där man hade samlat tre olika datatestverktyg i en liten behändig läderväska. Först ett korskopplingsfält för modemsignalerna, jämförbar med "blå lådan" som blivit oundgänglig för alla som arbetar med modem och datakommunikation. Den andra funktionen var för modemtest med testgenerator för sändning av bitmönster, för att testa datakommunikationen på analogsidan (telefonrättsidan). För att spara det bästa till sist, så hade man också lyckats få in en komplett linjelyssnare, med ett buffertminne på 1024 tecken. På en LCD-tablå kan man rulla fram och titta på 32 av dessa tecken. Man kunde även mata in ett testmeddelande som kunde sändas ut på linjeutgången. Modellen heter Trend Mini-Tester 100, priset uppgavs till 800 pund.

Har man mera pengar att spendera för testutrustning för datakommunikation hittar man i prisläget 3000 - 4000 pund en samling "intelligenta lådor", vars motsvarighet tidigare kostat över 100000:- sv. kronor.

Trend visade: "Trend DTA 1-30", Atlantic Research: "Interview 40A", Halcyon: "mini Fox 801A" och Spectron: "Datascope D101".

De senare kan till och med själva lista ut vad för sorts kommunikation som pågår på en ansluten datalinje. Man slipper alltså den tidsödande inställning av rattar och omkopplare för olika typer av dataöverföring.

Den andra utställningen var "All electronics/ECIF Show" på nyöppnade världiga Barbican Center i London.

Electronic Components Industry Federation stod som arrangör. Mest visades komponenter. Av intresse här kan nämnas att även Toshiba hade tecknat samarbetsavtal med Zilog för Z80 och Z8000 mikrodatorfamiljer. Intresserade kan läsa mera i Modern Elektronik nr.11/82.

Resans elektroniska höjdpunkt var utan tvivel "The Computer Fair" i Earls Court's gamla slitna utställningshallar.

Tillsammans med en enorm uppslutning av likasinnade vandrare jag runt och tittade på vad Storbritannien hade att erbjuda bland mindre datasystem. Kön framför ingångskassorna gick i flerdubbla led, nära en halv kilometer lång. Mest frapperande var att se den enorma dominans som Sinclairs ZX 81 uppvisade. Jag tror inte det fanns en monter där man inte hade en ZX 81 med ett eget "piratminne" eller egen utvecklad kringutrustningselektronik. Tänk att plocka ihop ett önskesystem med ZX81, 64K RAM och FORTH-system med några reläkort, digital/analog- och analog/digital-omvandlare för 200 pund!

Huvudnyheten var annars lanseringen av Sinclairs nya färgmaskin som fått namnet ZX Spectrum. Namnet hade valts för att betona att ZX 81 fortfarande var i produktion. Datatidningarna hade spekulerat om den nya datorn och kallat den ZX 82, men Sinclair framhöll att ZX Spectrum skulle ses som huvudkonkurrent till BBC's undervisningsdator. Någon tidningsreporter hade läxat upp Cleve Sinclair för att han hade sagt att ZX Spectrum var lika bra som BBC-datorn. Sinclair svarade: "Det har jag aldrig sagt, jag sa att den var bättre än BBC-datorn!"

Döm själv, här är lite data:

	ZX Spectrum	BBC Micro	VIC 20
Pris grundenhet	125	300	190
Standard RAM	16K	16K	5K
Maximum RAM	48K	32K	29K
Antal färger	8	8	16
d:o samtidigt pres.	8	4	16
Blinkande tecken	*	*	
Förhöjd ljusstyrka	*	*	
Höguppl. grafik	*	*	
Användardef. grafik	*	*	
Full ASCII 96 tn..	*	*	
Ljudgenerator	*	*	*
Skärm tn/rad*rader	32*24	40*25	22*23
Normal kassetbandsp.	*	*	
Överföringshast.Baud	1500	1200	300
LOAD & SAVE	*	*	*
VERIFY	*	*	*
MERGE	*	*	*

Högupplösningsgrafiken ger 192*256 pixels. BASIC-tolken ligger i 16 K ROM.

Skrivaren till ZX 81 passar också till ZX Spectrum.

Snart kommer ett "RS232/network interface board". För omkring 20 pund kan man ansluta ZX Spectrum till modem, serieskrivare eller andra ZX Spectrum i ett nätverk. Då får man även tillgång till Viewdatakompatibel programvara med 40 tecken/rad.

Man kommer också att lancera "ZX Microdrive". I Sinclairs broschyr talar man om en utbytbar microfloppy med lagringskapacitet på 100 K tecken och överföringshastighet 16 K tecken/sekund. Upp till 8 st. ZX Microdrives kan anslutas till en ZX Spectrum. Priset förväntas ligga runt 50 pund/st!

Priset för fullt utbyggd 48k-maskin är i England 175 pund!

Den enda ABC 80-maskinen på utställningen var försedd med en verklig "godbit"! Ett ungerskt företag hade utvecklat en mikrodiskett som man anslutit till ABC 80. Priset uppgavs till 300 pund (ca 3300:-kr.). För det fick man en låda med två drivenheter. Själva mikrodisketten bestod av en 3" enkelsidig flexskiva, helt innesluten i en skyddande plastkassett ungefär lika stor som en vanlig ljudkassett. Lagringskapaciteten uppgavs till 150 K tecken och överföringshastigheten 250 Kbits/sek. Accesstiden är jämförbar med vanlig flexskiva. Jag fick tillfälle att provköra systemet och det verkade som man fått full kompatibilitet med standard ABC 80 - DOS. Måntro vad Dataindustrier säger?

När jag körde LIB-programmet visade det sig att DR0: innehöll 384 sektorer medan DR1: innehöll hela 456 sektorer.

Slutligen bad man ABC-klubbens medlemmar vara välkomna till Datamässan i Älvsjö, utanför Stockholm i månadsskiftet september-oktober 1982. Där skall man vara med och visa sina produkter.

Göran Sundqvist
(tel.: 0756/303 10)

Mera om TV-editor

Sedan artikeln om TV-editor publicerades i ABC-bladet 1982:1 har vi hört TV-editor lovordas av många entusiastiska användare och vi har även fått flera brev med påpekande och frågor. På ABC-kassett nr 4 ligger en ny version av TVMAIN kallad TVMAIN3. Den innehåller en del nya kommandon och har fått några tidigare otympligheter borttagna. De viktigaste nyheterna är:

Negativa argument kan användas i vissa kommandon

Makrokommando

Kontroll att Q-registret inte spiller över Återinträde med LIST fungerar även med checksummor 10042 och 9913

De olika versionerna

På ABC-kassetter har distribuerats versionerna TVMAIN, TVMAIN2 och TVMAIN3. Behövs flera versioner?

De olika versionerna skiljer sig åt beträffande tillgängliga kommandon och buffertstorlek. Vill du jämföra de olika versionerna kan du skriva ut dem på skärm eller printer genom att först ta bort eller hellre ersätta med REM-sats rad 10 samt rad 11 i TVMAIN3.

Ta hänsyn till buffertstorleken

TV-editor är så konstruerad att allt tillgängligt minnesutrymme som inte behövs för programmet kommer att utnyttjas som textbuffert. Det är naturligtvis önskvärt att ha en så stor buffert som möjligt. Buffertstorleken bestämmer hur stor text som kan behandlas. Vill man skriva, läsa eller bearbeta en text som innehåller fler tecken än buffertstorleken måste man dela upp texten i två eller flera delar och lagra varje del som en separat fil.

I TV-editor är variabeln M\$ textbuffer, denna variabel har dimensionerats till det värde som variabeln M9% har. Det är lätt att ta reda på hur stor buffert man med olika versioner av TVMAIN får tillgång till med den utrustning som användes genom att stoppa programmet (med CTRL-C eller kommandot ;H) och sedan skriva ut värdet genom att skriva PRINT M9% <RETURN>. Vill man se hur mycket text som finns i bufferten gör man sedan återinträde med LIST (se nedan) och ger kommandot Z=. Skillnaden mellan M9% och detta värde är det utrymme man har kvar för ytterliggare text.

Beroende på den textbearbetning man vill utföra och den utrustning man har kan man behöva kompromissa mellan buffertstorlek och tillgång till nyttiga kommandon. Har du 16 K extraminne så kan du använda TVMAIN3 i de fall texten inte överstiger ca 70 sektorer. Varje radslut <CRLF> räknas som två tecken i bufferten. Av dessa lagras enbart <CR> vid skrivning på fil.

Buffertstorleken påverkas av Q-registrets maximala kapacitet. I Q-registret kan lagras in max 2000 tecken. Detta värde kan ändras genom att dimensionera om strängvariabeln Q\$ i rad 1010 till annat värde. Gör även motsvarande ändring på rad 2696 i TVMAIN3. Observera att när du gjort någon ändring i TVMAIN så går det inte att återinträda med list. Glöm inte att göra SAVE om du vill behålla ändringen.

Laddning av TV-editor

Det finns de som misslyckats med att ladda TV-editor. På ABC-kassett nr 2 låg ursprungsversionen av TV-editor, filerna TV.BAC, TVSUBR.ABS och TVMAIN.BAC. De kunde bara användas om man hade 5 tums flexskiveutrustning. Laddningen startas med RUN TV. Programmet TV består av raderna 1000 till 1060, resten är enbart REM-satser.

```
1000 POKE 65408%,33%,146%,255%,205%,
    94%,109%,17%,240%,255%,205%,27%,
    6%,56%,4%,33%,255%,255%,201%,
    33%,0%,0%,201%
1010 POKE 65520%,84%,86%,83%,85%,66%,
    82%,32%,32%,65%,66%,83%
1020 IF CALL(65408%) THEN 1040
1030 ; "Can't find TVSUBR" : STOP
1040 ; CHR$(12%);"-wait-"
1050 ; CUR(10%,0%);"Reenter with LIST"
1060 CHAIN "tvmain"
```

Tittar man närmare på de siffrvärden som pukas i raden 1010 så ser man att de motsvarar namnet på filen, "TVSUBR.ABS". Rad 1000 pokar in en liten rutin i maskinkod som sedan anropas i rad 1020. Denna rutin letar efter filen TVSUBR.ABS och läser från filen de maskinkodsrutiner som sedan utnyttjas av TV-editor. Rutinerna lagras i DOS-buffert 3-5, dvs fr o m adress 63488.

Christer Johnson <681> har påpekat att denna inlagring inte fungerade på hans DataDisc 82 som hade ett DOS märkt "6-81". När han ändrade 5:te byten i rad 1000 (rad 300 i TV80) från värdet 94% till 89% fick han laddningen att fungera.

Om laddningen inte fungerar skrivs texten "Can't find TVSUBR" ut på skärmen och programmet stoppar. Filen TVSUBR.ABS måste finnas tillgänglig i någon av drivarna. Därefter sker i rad 1060 chain till TVMAIN som också måste finnas tillgänglig. Lämpligen döper du den version av TVMAIN som du vill använda till TVMAIN.BAC och lägger den på den programskiva du använder för TV-editor.

Använder du enbart kassettutrustning så skall programmet TVCAS.BAC som låg på ABC-kassett nr 3 användas i stället för TV.BAC. I kassettversionen läses TV-editor's maskinkodsrutiner från filen TVSUB.BAC som alltså ersätter TVSUBR.ABS. Lämpligen gör du i ordning en systemkassett för TV-editor där du i tur och ordning lägger filerna TVCAS.BAC, TVSUB.BAC samt TVMAIN.BAC. Det sista programmet skall vara den version av TVMAIN du normalt vill använda, i förekommande fall omdöpt.

TVCAS och TVSUB kan som nämndes i föregående artikel användas även för flexskiveutrustning. Har du 80 teckens bildskärm (MyAB:s eller GeJo:s) skall du använda TV80.BAC, på ABC-kassett nr 3, i stället för TV.BAC. Även TVCAS fungerar för 80 teckens bildskärm.

Om du vill ladda in TV-editor igen efter att ha kört ett annat program, räcker det normalt att enbart ladda in TVMAIN på nytt. Maskinkodrutinerna ligger oftast kvar oförändrade. Samma gäller om du vill övergå till en annan version av TVMAIN.

Negativa argument

I TVMAIN3 kan man ha negativa argument till kommandona C, D, L, K och Q. Exempel:

-15C flyttar markören 15 steg bakåt i texten. -d tar bort det tecken som föregår markören, -2d tar bort bort två tecken, användes lämpligen när man markören står på början av en rad och man vill ta bort radslutet <CRLF> på föregående rad. Sekvensen -2DI<mellanslag> lägger även till ett space efter sista ordet på föregående rad. -20000D tar snabbt och effektivt bort allt från textens början fram till markören. -3L flyttar markören till början av rad 3 rader högre. Observera att om markören ej står på början av rad så inräknas även den rad som markören står på. -3K tar bort motsvarande rader/radbitar och -3Q lägger in dem i Q-registret. Kommandosekvensen -KK tar bort två rader om markören står på början av rad men bara en rad om markören befinner sig in på raden.

Du rekommenderas att innan du uppnått vana inte använda -3K eller -3Q förrän du först sett effekten av -3L. Glöm inte att flytta markören tillbaka igen efter testet, -3K tar bort 3 rader före det läge markören har när kommandot utföres.

Makrokommando

I TVMAIN3 har man tillgång till makrokommando. Som nämndes i artikeln i ABC-bladet 1982:1 kan man foga flera kommandon efter varandra som utföres först när man avslutar sekvensen med två högerpilar i följd. När man behöver samma sekvens flera gånger kan det vara praktiskt att först definiera sekvensen som ett makrokommando som sedan utföres varje gång man ger ett särskilt exekveringskommando. Kommandot M följt av den aktuella sekvensen av kommandon och avslutat med två högerpilar definierar sekvensen som ett makrokommando. Därvid inlagras sekvensen i det s k M-registret, variabeln M4\$, max 60 tecken. Där ligger det kvar tills man definierar en ny makrosekvens med nytt kommando M. Makrokommandot exekveras genom kommandot ;M (semikolon M).

En mycket kraftfull (och potentiellt farlig) finess är att makrokommandot kan göras rekursivt genom att den kommandosekvens man definierar som makro får avslutas med ;M. När vid exekveringen av makrokommandot ;M påträffas så ges nytt exekveringskommando och utförandet av kommandosträngen i makrokommandot upprepas ända tills något stoppvillkor blir uppfyllt. Stoppvillkor kan vara att någon sökning misslyckas eller markören vill flyttas utanför textens gränser.

Q-registret spiller över

När man i TVMAIN2 försöker lägga in för mycket text i Q-registret (mer än 2000 tecken) så spiller texten över i bufferten med följd att delar av den tidigare texten där blir ersatt med ny text som spillts över från Q-registret.

I TVMAIN3 har lagts in en kontroll som ger felutskrift när man försöker lägga in mer än 2000 tecken i Q-registret. Kontrollen görs i IF-satsen i andra delen av rad 2696. För att eliminera felet i TVMAIN2 kan man lägga in samma rad där.

Återinträde med LIST

Som förutskickades i förra artikeln har i TVMAIN3 lagts in en rad 11 som vid checksummor 10042 och 9913 ger samma funktion som rad 10 vid checksumma 11273. Du kan ta bort den rad du inte behöver men om du växlar mellan tangentbord med olika checksummor kan det vara bra att ha den kvar. Har du kvar rad 10 och har checksumma 10042 eller 9913 så kan du göra återinträdet med LIST 11 (eller ED 11). Prova! Många har undrat hur rad 10 egentligen ser ut, den går ju inte att lista. Svaret är att det finns ingen rad skriven i BASIC som kompilerar till den kompillerade formen som återgavs i artikeln i ABC-bladet 1982:1. Därför kan programmet inte lagras i BAS-format.

Programmakarna har utnyttjat en "feature" i BASIC-tolken som gör att återlistningsrutinerna i BASIC-en initierar exekvering av del av raden tolkad som maskinkod. Den i detta sammanhang väsentliga funktionen är att hopp sker till adress 3425 (13:97) resp 3423 (13:95) vid checksumma 11273 resp de båda andra. (Adress 3423 motsvarar 13:95. Ett litet tryckfel har här insmugit sig föregående artikel). Inhopp på adressen i fråga ger RUN-mode utan föregående CLEAR (nollställning av alla variabler). Rad 1000 i TVMAIN är också väsentlig i detta sammanhang, raden medför att raderna 1010- 1190 bara genomlöpes första gången vilket är väsentligt.

Rad 10 resp 11 kan extraheras, numreras om och sparas separat och genom MERGE fogas in i andra program som då även bör ha en rad motsvarande rad 1000 för att man skall få funktionen återinträde utan nollställning av variabler.

Finns det plats på disken?

När du arbetar med flexskiveutrustning med en text som du vill skriva på diskett kanske du ställs inför frågan om det du skrivit ryms på disketten. Det kan vara bra att veta hur många sektorer som finns lediga innan man skriver filen. Här kan programsegment TVLIB.MRG som ligger på ABC-kassetten nr 4 vara till hjälp. Detta segment är avsett att genom MERGE komplettera TVMAIN3. Faciliteten kostar dig 688 tecken i minskad bufferstorlek.

Vill du ha denna möjlighet förfar du så här: Gör LOAD TVMAIN3, sedan MERGE TVLIB.MRG och därefter SAVE TVMAIN3B. TVMAIN3B är en version som ger dig tillgång till två ytterliggare kommandon:

;0 skriver ut hur många sektorer som är lediga på drive 0,
;1 ger dig motsvarande för drive 1.

Kommandotabell

På mångas begäran publiceras en kommandotabell som sammanfattar alla kommandon som (hitills) finns i TV-editorn. Tabellen är ej avsedd att vara en fullständig beskrivning av tillgängliga kommandon utan mera som en minneslista eller lathund att ha lättillgänglig när man arbetar med TV-editorn.

Kommandon i TV-editorn

ORDREH.

Tecknet * användes för att i kommandona nedan beteckna <högerpil>. Kommandona skrivs här med stor bokstav men kan lika gärna ges med liten bokstav.

n betecknar talvärde som användes som argument vid vissa kommandon.
z betecknar antalet tecken i bufferten (användes som argument)
. anger markörens position räknat från textens början.

Direkta kommandon som utföres omedelbart för att flytta markören och visa olika delar av texten i bufferten:

<RETURN>	flyttar markören till början av nästa rad
<tskyt Y>	flyttar markören till början av raden resp till början av föregående rad
<högerpil>	flyttar markören ett steg åt höger
<vänsterpil>	flyttar markören ett steg åt vänster
<mellanslag>	visar den del av texten som står efter --more--
CTRL-B	flyttar markören till mitten på den sida som visas
CTRL-L	centrerar texten på skärmen symmetriskt kring markören

Övriga kommandon som utföres först när man tryckt tangenten <högerpil> två gånger i följd:

K<text>*	insättning av <text> på markörens plats
J	flyttar markören till textens början
nC	flyttar markören n steg åt höger (åt vänster om n är negativt)
zC	flyttar markören till textens slut
nL	flyttar markören till början av raden n rader längre ned (längre upp om n är negativt)
nS<text>*	söker efter n:te förekomsten av <text>
.=	skriver ut markörens position
z=	skriver ut antalet tecken i bufferten
nD	tar bort (delete) n tecken efter markören (före om n är negativt)
zD	tar bort allt efter markören
JZD	tömmer hela bufferten
nK	tar bort (kill) n rader efter markören (före om n är negativt)
nR<gammatext>*<nytext>*	byter ut (replace) första n påträffade <gammatext> mot <nytext>
nI*	lägger in tecken med ASCII-värdet n på markörens plats
nQ	lägger in i Q-register n rader efter markören (före om n är negativt)
G	sätter in texten i Q-registret på markörens plats
;Y	läs in text från fil och lägg den efter befintlig text i bufferten
;S	skriv innehållet i bufferten på fil
;X	skriv innehållet i bufferten på fil och exit från TV-editorn
;H	stopp
M<kommandosträng>*	definiera <kommandosträng> som makrokommando
;M	utför makrokommandot
;0	visa antalet lediga sektorer på drive 0
;1	visa antalet lediga sektorer på drive 1

Om man försöker utföra ett kommando som inte finns implementerat i aktuell version av TVMAIN kommer texten Illegal com: <kommando>

Kontrolltecken i texter

S k kontrolltecken, dvs tecken med ASCII-värde mindre än 32, ibland kallade "icke skrivbara tecken", användes för att styra olika printerfunktioner. De vanligaste är ASCII 13 <CR> carriage return eller vagnretur, ASCII 10 <LF> linefeed eller ny rad, ASCII 12 <FF> formfeed eller frammatning till ny sida, ASCII 9 <HT> horisontal tabulator, ASCII 8 <BS> backspace eller bakåttagning. Det finns printertyper som använder även andra kontrolltecken än dessa för att ställa in printern eller styra speciella funktioner vid utskrift. Någon enhetlig standard tycks inte finnas.

Det finns kontrolltecken som ABC-80:s filhanteringssystem inte klarar av att lagra textfil. Behöver man sådana kontrolltecken måste man lägga in dessa medan texten finns i bufferten och därifrån skicka utskriften direkt till printern. Det går att lägga in alla kontrolltecken i text som finns i bufferten. Ett kontrolltecken t ex <HT> som inte kan läggas in med det vanliga insättningskommandot I<text><högerpil> kan enligt vad Lars Frej <2260> påpekat läggas in med kommandot 9I<högerpil>.

Gunnar Tidner <1306>

FORTH för ABC80

av Torbjörn Alm (medlem 116)

Jag har under 2 veckor haft möjlighet att prova den fig-FORTH för ABC-80, som framtagits av Robert Johnsen, Uppsala Universitet, och som ABC-klubben förvärvat rättigheterna till. Programvaran kommer på nästa ABC-kassett och ställs till medlemmarnas fria disposition.

Den har överträffat mina förväntningar, och den är en mycket fin produkt. Den är enkel att använda, och innehåller en del extra finesser, som nyttjar ABC-80:s speciella möjligheter. ABC-80 FORTH baserar sig på fig-FORTH, som har tagits fram av FORTH Interest Group, San Carlos för fri spridning. Att lägga in den i en viss dator är givetvis förbundet med en hel del arbete. Programmet är ca 7 k Byte långt. Antal labels är precis på gränsen till vad avancerad assembler för ABC-80 tål (>500).

FORTH är ett mycket eget språk, som uppfanns av Charles Moore 1973, när han skulle styra ett radio-teleskop med hjälp av en PDP-11, utrustad med teletype och magnetband, en föga avundsvärd uppgift. Man arbetade samtidigt med en annan dator, och enbart med assembler. Genom att skapa FORTH, som kan betecknas som ett maskinberoende (i stor utsträckning) lågnivåspråk, kunde man lösa sina problem. Sedan dess har språket spritt sig, inte minst tack vare tillkomsten av fig-FORTH.

Utöver fig-FORTH eller FORTH-79, finns det POLYFORTH, som säljs av FORTH Inc., som drivs av just Charles Moore. Det finns vissa olikheter mellan dessa dialekter, men i stort sett kan man flytta FORTH-program mellan olika maskiner. Det tycks finnas en tredje variant, MMSFORTH för TRS-80, men jag vet inte vad den liknar.

Vad är nu FORTH? Språket är ett "threaded language". Koden består av en följd av adresser till rutiner, som i sin tur kan vara i masinkod, s.k. primitiver eller i FORTH-kod, sekundära funktioner. En av ideerna med fig-FORTH är att 90% av koden är sekundära funktioner, och de kan direkt flyttas mellan olika processorer, under det att ca 10% är processorunik assemblerkod.

Ett FORTH-system innehåller bl.a. följande delar: ett user interface med rutiner för inmatning och utmatning till skärm och till skrivare, en interpreter, som exekverar FORTH-koden, en kompilator, som översätter FORTH källkod till internkod samt ett system för virtuellt yttre minne. Dessutom finns det en editor för att editera FORTH-program samt en assembler för att kunna skriva primitiva funktioner.

FORTH förutsätter enbart att det finns något yttre medium på vilket man kan läsa och skriva block. Dessa block slås sedan ihop till sk. screens, vilka brukar vara 1 k byte, men i ABC-80 har valts 768 bytes (3 block), då det är enklare att hantera på skärmen.

För att editera program har man i ABC-80 FORTH inte mindre än 3 olika editorer till sitt förfogande:

fig-FORTH editor, som ingår i fig-FORTH installationsmanual, och finns beskriven i denna.

POLYED, som är ursprungligen definierats för POLYFORTH, och som finns beskriven i Leo Brodies bok: Starting Forth.

VEDIT, som är en enkel och lätthanterlig bildskärmseditor.

Varför är FORTH så populärt?

Språket är mycket maskinnära utan att vara masinkod, och programmen blir mycket effektiva. Det andra är att det är ett rent strukturerat språk. Någon GOTO finns inte, utan man arbetar med funktioner, sk. FORTH-ord, vilka i sin tur byggs upp av egna ord eller ord i systemet. Det finns LOOP-funktioner, IF-THEN-ELSE m.m. och det är enkelt att skapa en CASE-sats, vilket jag visar senare. En mycket trevlig egenskap är att man kan skriva in en funktion på skärmen, ladda funktionen i kompilatorn och sedan omedelbart testa den, samt om den gick fel, editera och testa igen. Det gäller även FORTH-funktioner i assembler. Man kan jämföra med arbetet med program i assembler!

Efter hand får man ett bibliotek med screens med fungerande rutiner, som sedan kan kombineras ihop. Alla funktioner anropas med sitt namn, och enda kravet vid kompilering av en funktion är, att de refererade funktionerna är definierade.

Man kan definiera konstanter och variabler med namn, och man kan ha konstanter direkt i koden. Det går även att beräkna adresser till funktioner och lägga dem på stacken, för att sedan exekvera dem.

Den assembler, som finns till FORTH ABC-80 kan assemblera 8080-kod. För den som vill ha en fullständig Z80-assembler, rekommenderas Loeligers bok: Threaded Interpretative Languages.

Start av FORTH:

FORTH ABC-80 är ett .ABS-program, och startas över CMDINT.

1. Sätt FORTH system-skiva med FORTH-screen och FORTH.ABS i drive 0.
2. Gör BYE. När CMDINT laddats, skriv FORTH.
3. När FORTH laddats, svarar den med att överst på skärmen skriva Z80 FORTH I.1
4. Skriv 6 LOAD 10 LOAD RETURN . Nu laddas en del användbara funktioner. Bland annat laddas funktionen WHERE, som kan användas för att söka fel vid inläsning av program samt laddrutiner för fig-FORTH editor, POLYED samt assembler. Screen 10 innehåller programmet PRINT-BLOCKS, med vars hjälp screens kan skrivas ut på skrivare, 2 per sida.
5. Om man har diskar med enkel densitet, skriver man 0 DENSITY ! RETURN.
6. Nu är det klart att köra.

Skärmeditor VEDIT.

För att använda VEDIT och editera screen 70, skriver man 70 VEDIT. Screenen listas på skärmen med radnummer i vänsterkant och rutor i högerkant. Om man skriver på tangentbordet, läggs tecknen in där markören befinner sig. Om man vill förflytta markören använder man pilarna för förflyttning åt vänster eller höger, RETURN för att gå nedåt och ü för att gå uppåt, på samma sätt som i TV-editorn. Om man gör CTRL/E' lämnar man VEDIT, och för att spara screenen på disken gör man FLUSH RETURN.

VEDIT kan inte användas för att spränga in text i eller mellan rader. För detta behöver man tex. POLYED. För att ladda POLYED skriver man POLYED RETURN efter det att man gjort 6 LOAD. För att sedan aktivera editorn, skriver man EDITOR RETURN. Då kommer de speciella editeringskommandona att aktiveras. Därefter skriver man 70 LIST RETURN. Som synes skall varje kommando eller följd av kommandon avslutas med att man gör RETURN för att de skall exekveras. När alla kommandon är klara, skriver FORTH ut ok som kvittens och är klar för nästa kommando. Det gäller även editorerna.

POLYED-kommandon:

WIPE blankar aktuell screen.

L listar aktuell screen.

5 T skriver ut rad 5 och pekar ut 5 som aktuell rad.

U TEXT lägger in TEXT under aktuell rad. Underliggande rader flyttas ned. Rad 20 försvinner.

P TEXT ersätter aktuell rad med TEXT.

F XXX söker från aktuell rad efter XXX. Vid träff skrivs raden med XXX ut, och markören står efter XXX.

R YYY ersätter det som stod i söksträngen. (XXX) med YYY.

E tar bort XXX.

FLUSH återskriver screenen till disken.

Det finns ytterligare kommandon, som jag ännu inte hunnit stifta bekantskap med. POLYED finns beskriven i Brodies bok Starting Forth.

Hantering av printer och V24-utgång.

ABC-80 FORTH innehåller dels en V24-handler, som i princip är standard V24-handler, och som kan mata ut vid valfri baudrate, dels länkning till parallell-printerkoden, som skall finnas i 7800H, dvs Centronics snitt eller P40. Det finns en intern variabel, PR-TYPE, som normalt är =1, dvs Centronics, men om man skriver 2 PR-TYPE !, avses V24. Baudrate anges med kommandot 2400 BAUD. Om man sedan sätter printerflaggan genom att skriva PR-ON, kommer att utmatning till skärmen även att gå till printer. Dessutom finns det 2 funktioner, ASCII-IN och ASCII-OUT, vilka kommunicerar via V24-utgång, oberoende av printer-flaggor.

Exempel på FORTH-verb.

FORTH ABC-80 innehåller över 200 kommandon, och skulle jag beskriva dem alla, blev det en hel bok. Eftersom det finns flera bra sådana, nöjer jag mig några få:

+ Addera de två översta talen på stacken, och lägg tillbaka resultatet. Likheter med en HP-kalkylator är slående. På samma sätt fungerar ***** och **/**.

. trycker talet på toppen av stacken.

/MOD utför en heltalsdivision och lägger tillbaka såväl kvot som rest.

/* gör först en multiplikation, sedan omedelbart en division. Vitsen ligger i att mellanresultatet hålls i dubbel precision.

£ Hämtar ett 16-bits ord, vars adress finns på stacken.

! lagrar ett 16-bitars ord på stacken enligt en adress på stacken.

: NAMN <instruktioner>; definierar funktionen NAMN och lägger in den i ordlistan.

' (kallad tick) används för att lagra adressen till en funktion.

' ABC CFA lägger adressen till ABC på stacken. Om man sedan gör EXECUTE, exekveras ABC. Här finns en skillnad mot POLYFORTH, som inte framgår av Brodies bok.

Användbara hjälpfunktioner i ABC-80 FORTH:

VLIST listar namnet på alla funktioner som hittills definierats. Om du gjort PR-ON så erhålls lista även på printer.

3 30 INDEX

listar första raden på skärmarna 3-30. En god regel är att första raden (rad 0) på varje screen skall innehålla en kommentartext om screenen. Kommentartext inleds med (och avslutas med).

3 30 PRINT-BLOCKS

(finns på screen 10) listar screen 3-30 på printer, 2 screens per sida.

CTRL/< RETURN

dumpar skärmen på printer. Denna finns på screen 8 och laddas av 6.

Trace av FORTH-kod.

Denna funktion finns inte med på kassetten, men har visat sig mycket bra. Om man lägger upp den på en screen och läser in den före de rutiner man vill spåra, erhåller man en utskrift med namn och toppen av stacken för varje rutin, om variabeln TFLAG sätts =1.

```
0 ( TRACE )
1 FORTH DEFINITIONS
2 0 VARIABLE TFLAG
3 : (TRACE) ( IF TFLAG IS ZERO, THEN
4 NO TRACE OCCURS )
5 TFLAG E'E IF CR R 2-NFA DUP ID.
6 CE'E 31 AND 32 SWAP - SPACES
7 -2 4 DO SPE'E I+E' 8 .R-2 +LOOP
8 THEN ;
9 : : ( REDEFINE : )
10 ?EXEC !CSP CURRENT E' CONTEXT!
11 CREATE ' (TRACE) CFA DUP E'E
12 HERE 2 - ! , RB ;
13 ;S
```

Funktionen visar vad som kan göras med FORTH.

Ett annat exempel är definitionen av en CASE-sats.

```
( CASE )
FORTH DEFINITIONS DECIMAL
: CASE:
<BUILDS SMUDGE RB
DOES> SWAP 2 * + E'E EXECUTE
;
;S
```

```
( EXAMPLE )
: OPET ." APA" ;
: LPET ." BABIAN" ;
: 2PET ." ANNAT" ;
CASE: DJUR OPET LPET 2PET ;
0 DJUR ( SKRIV APA )
1 DJUR ( SKRIV BABIAN )
2 DJUR ( SKRIV ANNAT )
```

CASE: är exempel på en högnivå-definition, som är en avancerad feature hos FORTH. Det är ingen tvekan om att FORTH är ett språk, som sätter fantasin i rörelse hos alla hackers. Det finns nästan outtömliga möjligheter att skriva kompakt kod. Tyvärr tenderar den lätt att bli write-only, dvs. ingen, ej ens författaren kan läsa den.

FORTH-Litteratur:

1. **Starting FORTH**, Leo Brodie, Forth Inc. Prentice-Hall. Detta anses allmänt, och det med rätta, vara den absolut bästa introduktionen till FORTH. Boken är lättläst och har humor. Den är rikt illustrerad av författaren. Här finns POLYED beskriven.

2. **fig-FORTH Installation Manual**. Forth Interest Group, San Carlos. Här finns dels stora delar av interpretaren i FORTH-kod, så att man sätta sig in i vad den har för sig. Här finns även beskrivningen av fig-FORTH editorn.

3. **ABC-klubben RAPPORT 2, Fig-FORTH on ABC-80** implemented by Robert Johnsen. Juni 1982.

En rapport 46 sidor skriven på engelska och innehållande bl a installationsmanual för ABC-80 med köranvisningar och screen-beskrivningar samt utförlig referenslista. Dessutom innehåller rapporten utdrag ur ref 2 i form av Glossary (verb-beskrivningar) och beskrivning av Fig-FORTH editorn.

Här är Fig-FORTH!

Fig-FORTH som presenterades i ABC-bladet 1.5 (sommarutgåvan) har utvecklats av docent Robert Johnsen vid Fysikalisk-kemiska Institutionen, Uppsala universitet. Den kan köras på en standard ABC-80 utrustad med 5-tums flexskiveutrustning. Programmet är skrivet i assembler och köres under CMDINT.SYS.

Torbjörn Alm har provat programvaran och skrev en recension för ABC-bladet 1.5 som p g a ombrytningsfel återges på nytt.

Programvaran distribueras på kassett till ABC-klubbens medlemmar. På ABC-kassett nr 4 ligger FORTH16.ABS för ABC-80 utan extra minne. Den kan köras även på ABC-80 med 32 K RAM men därvid utnyttjas inte hela RAM-minnet effektivt. För den utbyggda versionen finns FORTH32.ABS som läggs på ABC-kassett nr 5. FORTH utnyttjar inte det vanliga filhanteringssystemet på ABC-80 utan lagrar program och data på sk screens. Varje screen upptar 3 sektorer på disketten och lagras fr o m spår 3 sektor 1 och uppåt. Man bestämmer själv hur stort utrymme man vill reservera för screens. Om man reserverar plats för 75 screens så ryms hela programbibloteket, tiotalet egna screens samt FORTH.ABS och CMDINT.SYS på en skiva med enkel densitet. Programbibloteket till FORTH distribueras i form av textfilerna SCREEN.TXT och SCREEN2.TXT. SCREEN.TXT innehåller screens nr 3 till 36. SCREEN2.TXT som läggs på ABC-kassett nr 5 innehåller screens nr 34 till 69 samt en fullständigare version av screen nr 7. Screens nr 34 till 36 i SCREEN.TXT kommer att skrivas över av andra screens från SCREEN2.TXT. Det kan vara skäl att spara screen 36 genom att flytta den till annan plats med nummer högre än 69.

Fortsättning på sidan 23

4. **Threaded Interpretative Languages**, R. G. Loeliger. Byte Books (McGraw-Hill). Denna bok beskriver på maskinnära nivå hur en FORTH-interpretar byggs upp av ett stort antal små rutiner och hur de samverkar. Här finns också en Z80-assembler i FORTH, som bör gå att göra om till ABC-80 FORTH. Den innehåller mycket få fel. Jag har själv implementerat denna FORTH, och fann 2 fel! Den kan dock inte nyttjas till produktion, eftersom den saknar väsentliga delar. För den som vill veta mer om hur FORTH arbetar internt, rekommenderas denna bok varmt.

5. **FORTH DIMENSIONS** är en tidskrift, som ges ut av Forth Interest Group. Den innehåller artiklar och FORTH-program. Jag vet inget om denna själv, men Robert Johnsen, som gjort ABC-80 FORTH, har hämtat en hel del godbitar ur denna, varigenom ABC-80 FORTH är bättre än standard fig-FORTH på flera punkter.

ABC80 blir RAM-maskin

Jag har låtit bygga om min ABC80 till RAM-maskin. Jag fick hjälp av en kompis som är snitsare. Den ombyggnadssats jag valde var Ge-Jo's. Ge-Jo står för Göran Johansson i Ljung, en kille som gjort en hel del intressanta saker för ABC80. Nåväl nu gällde det att montera in en RAM-tillsats, vilket visade sig betydligt svårare än att stoppa in 16K expansionsminne. Det är stor skillnad mellan bygganvisningarna till RAM-tillsats och expansionsminne. Byggskrivningen för expansionsminnet är klart bättre. En del punkter i beskrivningen skulle nog delas upp i flera. En ytterligare komplikation är att beskrivningen inte stämmer för alla tangentbord. Man letar alltså förgäves efter en viss kondensator om man har ett äldre tangentbord, men i sannings namn ska sägas att Göran Johansson är en hyggelig kille som gärna svarar på frågor. Varför ska man nu bygga om sin ABC80 till RAM-maskin? Jag kan se flera skäl som kan delas upp i två huvudgrupper. Ett skäl kan vara att man vill ha möjlighet att utnyttja något annat operativsystem exempelvis CP/M. Ett annat skäl kan vara att göra förändringar i Basic-tolken. För de som undrar hur en ABC80 fungerar efter en ombyggnad ska jag försöka förklara.

Det är möjligt att köra sin ABC80 ungefär som vanligt efter en ombyggnad. En ombyggd ABC80 fungerar dock annorlunda eftersom RAM-minnet är tomt när strömmen slås på måste någonting hända, som kopierar Basic-tolken från PROM-kapslarna till RAM-et. Detta sker också automatiskt, varefter man kan köra som vanligt. Det är inte bara Basic som kopieras utan även printer-program och den del av Basic som ligger på controler-kortet i floppyn.

Mer minne

Ombyggnaden leder till att man får 32K RAM-minne för program och data ovanför bildminnet. Under bildminnet får man nu RAM-minne inom ett adressområde som tidigare varit tomt. Om man inte har floppy och 40 kol skärm så får man alltså 32 K minne ovanför bildminnet och 15 K mellan Basic-tolken och bildminnet. Har man floppy så får man 8 K minne mellan diskfilhanteringsdelen av Basic-tolken, dessutom får man

3 K direkt under bildminnet. Vad ska man nu använda det nya minnet till? Ja när det gäller de 32 K som ligger över bildminnet så används dessa som vanligt. Man kan ha större program, deklarerar stora matriser osv. Det är alltså ingen skillnad mellan dessa 32 K och ett minne som expanderats till 32 K utan att maskinen ändrats för övrigt. Det som är spännande är alltså att utnyttja minnet under bildminnet och att utnyttja möjligheten att ändra i Basic-tolken. Det är då fråga om att utnyttja helt nya möjligheter som bara ges då man bygger om till RAM-maskin. Det nya minnet kan exempelvis användas för lagring av maskinspråk, men det är faktiskt möjligt att ladda in Basic-program under bildminnet. För att kunna göra detta måste man ändra BOFA till 16384. Den ändringen sker med kommandot 'POKE 65053,64'. Jag har exempelvis laddat in Ge-Jo's program Subrutin, varefter jag ändrat BOFA med 'POKE 65053,128'. Och det fungerar! Nästa steg är att lägga Lathund och ABCV24 mellan 16 K och 24 K.

Jag har ändrat promptern till K-Å J och i stället för 'ERR 21' får jag 'FEL 21'. Dessa förändringar åstadkommer jag genom att poka in resp text i stället för befintlig. Principen är densamma vid alla förändringar av de typer som nämnts ovan. Problemet består då endast i att hitta resp textsträngar i BASIC-tolken, vilket kan vara nog så svårt. Lämpliga hjälpmedel förutom tålamod är Disassemblerlistan och något program som letar efter textsträngar i BASIC-tolken. Detta program ska skriva ut första adress i resp sträng. När man skaffat sig en förteckning över var i tolken som RUN, GOTO, LIST osv ligger kan man börja ändra.

Att ändra sin BASIC

När Basicen ligger i RAM-minnet så kan man ändra i den på ett ganska enkelt sätt. Man använder sig antingen av POKE-kommandon (när det är små ändringar) eller program bestående av POKE-satser. Exempel på ändringar som man kan göra är: förändrad prompter, ändrade namn på kommandon och nya namn på Basic-satser.

BASIC på Svenska

En möjlighet som öppnar sig är att göra en BASIC på Svenska. Det gäller att översätta kommandon och satser till Svenska motsvarigheter. En komplikation är att de Svenska översättningarna måste bestå av

lika många tecken som de Engelska motsvarigheterna. Man kan tänka sig följande exempel på översättningar:
LET=GÖR, GOTO=HOPP, POKE=FYLL, PEEK=KIKA etc.

Vad kan man ha för nytta av Svensk BASIC?

Nytan är förmodligen inte så stor, men kanske en Svensk BASIC kan ha ett visst värde i undervisning. Framför allt bör det bli lättare att använda datorn på mellanstadiet i grundskolan. En ABC80 med endast RAM-minne som fått sin BASIC ändrad till Svenska, kan köra program lagrade i BAC-format som vanligt. Det förhåller sig annorlunda med BASIC-program i BAS-format. Ett BAS-program kan bara köras om det innehåller samma satser som BASIC-tolken. En Svensk BASIC kan alltså användas för kryptering. Gör så här, läs in det BAC-program som ska krypteras och spara det sedan med LIST. Nu innehåller den sparade filen dina översatta BASIC-satser.

En mera nyttig användning av en ändrad BASIC skulle kunna vara att göra program skrivna i andra BASIC-dialekter körbara på ABC80. Det är naturligtvis inte möjligt att ändra funktionen av en sats genom att byta stavning av den, men när två BASIC-dialekter skiljer sig åt genom att använda olika satser för samma funktion, kan det räcka med att ändra på det sätt som vi beskrivit.

Att ändra funktioner i BASIC

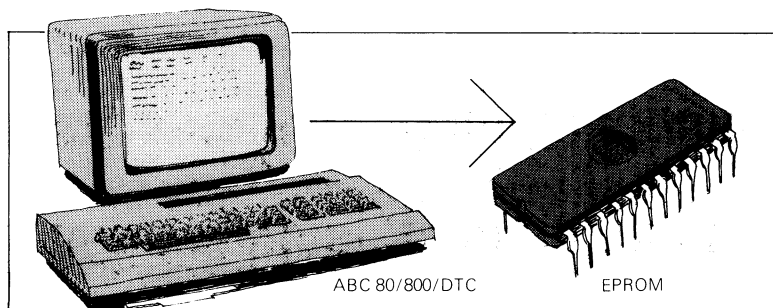
Jag har ännu inte kommit så långt att jag börjat ändra BASIC-tolkens funktioner och jag har inte heller försökt lägga till nya. Jag har däremot haft svårt att avhålla mig från att fundera över vad som skulle vara möjligt. En intressant sak skulle vara att bygga ut BASIC i utrymmet över 16 K. Det man måste göra är att beskriva de nya satsernas funktion, ge dem namn och operationskoder. Med operationskod avser jag den interna halvkompilerade kod, som ABC80 arbetar med internt.

DP/M och andra operativsystem

På denna front finns för närvarande inte så mycket att säga eftersom Göran Johansson inte är färdig med CP/M för sina ombyggda ABC80. Han räknar att bli färdig till hösten. För min del är det dock inte någon brådska - jag ska lära mig FORTH!

Kjell-Åke Johansson

Utveckla med ABC⁸⁰/₈₀₀ DTC!



Ring oss på SATTCO så får du veta mer om dina utvecklingsmöjligheter med DataBoard.

Använd din dator som utvecklingssystem för DataBoard 4680's kortserie!

- ☐ Skriv ditt applikationsprogram i Basic. Provkör ditt program och I/O-korten i ABC 80/800 DTC.
- ☐ Konvertera Basic-programmet med "Prommerbar Basic" till PROM.
- ☐ Överför programmet enkelt till EPROM med hjälp av prom-programmerare 5071.
- ☐ Enkortsdatoren med det framtagna programmet och I/O-kort ersätter ABC 80/800/DTC i t ex en mät- eller styr-applikation.



SATTCO

DATAAVDELNINGEN DALVÄGEN 10, 171 36 SOLNA TEL 08-734 00 40

PROGRAMMERARE

sökes för arbete på ABC 80/800 m fl.
Viktigare än "smartprogrammering" är systematisk uppbyggnad och provning, intresse för dokumentation och användarvänlighet. Vidareutveckling och nyframtagning av interaktiva dataprogram. Erfarenhet av film eller videoteknik är en fördel.

Profilm, startat 1970, utvecklar kvalificerad utrustning och metodik för filmlaboratorier och TV-bolag i Sverige och utlandet.

Mer information lämnas av:

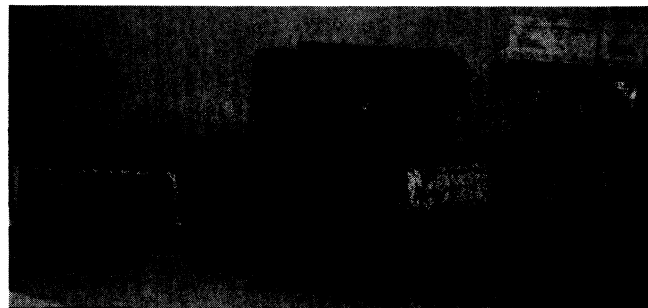
Tomas Dyfverman, tfn 08/83 60 05.

Eller skriv till:

Profilm Service AB
Östervägen 14 A,
171 39 Solna



ABC DAMMSKYDD



SKYDDA DIN INVESTERING

SKYDDEN ÄR TILLVERKADE I LJUSBRUN VINYLPLAST MED VÄVFÖRSTÄRKT INSIDA. MATERIAL ET ÄR MYCKET MJUKT, OCH ÄR DÄRFÖR LÄTT ATT VIK A IHOP. DÄ SKYDDEN ÄR VATTENTÄTA, GÄR DE MED FÖRDEL ATT TORKA MED FUKTIG TRASA. SKYDDEN ÄR ÄVEN ANTISTATISKA.

ABC80M	MONITOR.....	105:-
ABC80	DATOR.....	105:-
ABC800	DATOR.....	115:-
ABC800F	SKRIVMINNE.....	95:-
ABC810	FÄRGMCHITON.....	105:-
ABC815	MONITOR.....	137:-
ABCMx80	SKRIVARE.....	105:-

ALLA PRISER EXKL. MOMS.

SKYDD FINNS ÄVEN FÖR: APPLE, TRS 80, ATARI, CROMEMCO, SUPERBRAIN, HEATH COMPANY, OHIO SCIENTIFIC MED FLERA.

ÅTERFÖRSÄLJARE SÖKES!

M.T AUTOMATIK, BOX 4057, 182 04 ENEBYBERG, TEL: 08/768 83 90

DATASVENSSON



ANNONS



Vad är då R 81? Jo, Redovisning 81 på ABC 80. En utmärkt bokföring för den lille företagaren.

Redovisning 81 ingår i programserien Modul 81 och fås ifrån din närmaste ABC 80 handlare.

Finns bl.a. till diskarna: Luxor Flex, Metric FD2D, Micropolis, Datadisc 82. Utbyggt internminne behövs i datorn.

DATAKRAFT AB, BOX 3045, 22003 LUND. Tel. 046-137100.

Kassett ett

CASTODSK.BAS

Detta program kopierar en fil från kassett till floppy. Det kan alltså endast köras av den som har floppy. Vid inladdning från kassetten i system utan floppy, eller om floppyn ej är del av systemet, får man felmeddelandet ERR 8 LINE 380. Detta felmeddelande beror på att instruktionen på rad 380 ej kan tolkas av den Basic som finns i tangentbordet. Vid instruktionen NAME sker ett hopp till den utbyggnad av Basic som finns på controler-kortet i floppyn. Felmeddelandet ovan är alltså i sin ordning när systemet saknar floppy.

Vid körning av programmet bör man vara uppmärksam på att det kan förstöra filer på floppyn ifall dessa råkar ha samma namn som filer på kassetten, som ska kopieras. Mot detta skyddar man sig genom att alltid kopiera till tom floppy.

Tänk på att CASTODSK endast kopierar text-filer. Det innebär att program som sparats med SAVE ej kan kopieras med programmet. Det kan vara bättre att använda CASDISK som numera ligger som första fil på ABC-kassetterna.

MAXEN.BAS

Happening, rolig och enkel roar barn. Körs på grundutrustning liksom alla program tom OHMDATA.

SLALOM.BAS

Det gäller att ta sig fram mellan portarna. Du kan tävla med dig själv eller med dina vänner. Det fungerar bäst om tangentbordet är låst i versal-läge och röd lampa lyser.

REAKTION.BAS

Testa din reaktionsförmåga, du kan tävla med dina vänner.

GUBBEN.BAS

Gissa ord. Missar du blir du hängd.

KRIG.BAS

Krigsspel

MÖNSTER.BAS

Ritar (vackra?) mönster på skärmen. Bra exempel på vad grafiken kan - och inte kan!

LUFFAR.BAS

Luffarschackprogram

VIRRE.BAS

Underhållningsprogram av originellt slag. Virre är en man som rör sig på olika sätt. Virre kan hoppa tala och mycket annat som du kan styra. Virre är de små barnens förtjusning, men har många vuxna anhängare.

MYSKO.BAS

Är mycket mysco - grafik och ljud och överaskning! Elaka tungor förmåler att programmet innehåller en bug!

KULGRAF.BAS

Grafikprogram som ritar ett regelbundet mönster på skärmen. Mönstret tättnar tills skärmen är helt vit varvid mönstret börjar upplösas och försvinna osv i all oändlighet.

SLOOP.BAS

Musik ur ABC80 är det verkligen möjligt? Kan man verkligen få ut något annat än pang! choff! oioio! ur ABC80? Du får avgöra den saken själv genom att lyssna på Sloop.

EURFRÅG.BAS

Frågesport för familjen. Vem vet mest om Europa?

MATTEN.BAS

Program för att öva aritmetikfördigheter. Programmet är på mellanstadienivå och kan användas i skolan.

ROTEN.BAS

Kan köras på grundutrustning. Programmet beräknar kvadratroten av ett tal med valfritt antal decimaler, dock högst 20. Programmet är självinstruerande och lätt att använda, även för absoluta nybörjare.

PRIM.BAS

Program som delar upp ett tal i primfaktorer. Programmet är intressant eftersom det var en vanlig uppgift för de första datorerna. Programmet klarar av att dela upp tal med upp till 28 siffror. Det tar lång tid att dela upp stora tal.

LÅN.BAS

Hur fördelaktiga är dina lån?

STÅLTEMP.BAS

Du kan undersöka olika legerade ståls behov av förvärmning. Du kan variera stålets sammansättning dimension etc. Du kan variera innehåll av svavel, fosfor och kol samt flera legeringsmetaller.

OHMDATA.BAS

Undervisningsprogram i elektricitetslära. Du tränar begrepp och beräkningar t ex resistans, spänning och effekt.

ABCMINL.BAS

Kräver floppy och modem som extrautrustning dessutom krävs terminal-program ex ABCV24.BAS, vilket finns på ABC-kassett nr 3. Programmet medger mottagning av textfiler från klubbens monitor i Alvik och från liknande utrustning på andra ställen.

CASMINL.BAS

Kräver modem som extrautrustning. Programmet medger överföring av filer från klubbens monitor till din ABC80.

ABCTrans.BAS

Kräver floppy och modem som extrautrustning. Programmet som nu finns i en version för 80 kolumners skärm, medger filöverföring i båda riktningarna mellan din utrustning och klubbens monitor.

LOGAIN.BAS

Kommunikationsprogram som liknar ABC-TRANS och följaktligen kräver modem och floppy. Programmet är dock utformat med tanke på kommunikation med stordatorer av fabrikat DEC -Digital Equipment Corp. Programmet innehåller exempelvis en automatisk inloggningsrutin. Det som gör programmet intressant är att klubben snart kan erbjuda sina medlemmar möjlighet att utnyttja QZ's DEC10-dator till en starkt rabatterad taxa.

MAKETEXT.BAS

Kräver floppy som extrautrustning. Programmet är en mycket enkel editor. För den som inte vet vad en editor är kan jag berätta att en editor är ett program som gör det möjligt att skriva text på en fil. MAKETEXT kan användas för att skriva brev mata in data i en fil som kan bearbetas av ett annat program.

Du använder MAKETEXT så här:
Ladda in MAKETEXT och skriv RUN. zzz

VISA.BAS

Kräver floppy. Programmet gör det möjligt att på ett bekvämt sätt läsa textfiler rad för rad. Du kan se hur programmet fungerar om du gör så här. Du har VISA.BAS på en diskett! Skriv RUN VISA och ge när programmet svarar VISA.BAS som indata. Du kan nu läsa filen genom att trycka ned ex mellanslagstangenten.

FILSUM.BAS

Programmet beräknar antalet rader och ASCII-summan av tecknen i en textfil på skiva. Det är ofarligt att använda, det är alltså fritt fram för experiment.

FILCOM.BAS

Filhantering på floppy. Programmet jämför två filer. Programmet är ofarligt och kan ej skada några filer. Förutsättningen för att jämförelse ska kunna ske är att filerna är text-filer, vilket innebär att programmet måste sparas med LIST för att kunna jämföras. Du kan prova FILCOM på följande sätt. Ladda in FILCOM med LOAD och spara sedan med LIST XYZ. Du får då en version av programmet under namnet XYZ.BAS. Du kan nu köra FILCOM, skriv RUN och ge FILCOM.BAS respo XYZ.BAS som indata till programmet. Om filerna är lika så får du meddelande om detta. Skulle det vara någon skillnad skrivs de rader som är olika ut.

CONCAT.BAS

Kräver floppy som extra utrustning. Programmet konkatenerar filer, dvs slår samman två eller flera filer till en. Det kan du ha användning för om du skrivit text med MAKETEXT och vill lägga två text-filer efter varandra i samma fil.

Varning!

Tänk på att det namn du ger din utfil måste vara skilt från det namn dina in-filer har. Är namnet du ger utfilen upptaget förstörs den filens innehåll.

SORTCORE.BAS

SORTCORE sorterar en textfil internt i RAM-minnet. Programmet är skrivet för flexskiva men fungerar även för kassett. Först frågar programmet efter infil. Se till att den textfil som skall sorteras ligger klar att läsas på flexskiva eller kassett. Därefter frågas efter recordlängd, d v s den maximala radlängden som posterna i infilen har. Sedan frågas efter "sorteringsnyckel(start-position,längd)". Svara då med två tal separerade med komma. Det första talet skall vara första position i det fält man vill ha raderna sorterade efter och det andra talet fältets längd.

Exempel:ett adressregister där varje rad har fyra fält, namn (efternamn förnamn), gatuadress, postnummer, postadress, med fältlängderna 20, 25, 5 resp 13. Recordlängden blir då 63. Vill man sortera alfabetiskt efter efternamn anger man sorteringsnyckel 1,20 och vill man ha det i postnummerordning anger man nyckeln 46,5.

Programmet talar om hur många records, d v s poster som kan sorteras. Får du ERR 3 så minskas värdet på N% i rad 30 och försök igen.

Programmet frågar sedan efter utfilens namn. Trycker man bara RETURN får utfilen samma namn som infilen.

Kassett två

När textfilen lästs in från flexskiva (eller kassett) stannar programmet och väntar på tangentnedtryckning. Om man vill kan man nu passa på att byta skiva (bra när utfilen fått samma namn som infilen och man inte vill få den överskriven). Använder man kassett så gör man nu bandspelaren klar för inspelning innan man trycker tangent. Därefter startar sorteringen. När den är klar skrivs sorteringsstatistik ut och den sorterade utfilen skrivs på flexskiva resp kassett. När det är klart skrivs utfilens namn ut.

KONVERT.BAS

Program som konverterar vissa tecken i en fil. Programmet verkar föga användbart men visar hur man kan lösa ett problem, som är generellt.

DÖP.BAS

Programmet används för att märka skivor med volym-namn.

IDMÄRK.BAS

Märker eller 'döper' en floppy. Liknar alltså programmet DÖP, men kan dessutom kolla eventuell märkning av skivan.

CKOLON.BAS

Lib-program för floppy. Programmet är skrivet i maskinspråk och mycket snabbt. Du startar programmet med RUN CKOLON varvid programmet går igång och snabbt ger Libben plus resterande utrymme på floppyn i drive 0 och 1. Har man många filer på drive 1 så att filerna på drive 0 hinner rulla förbi kan man ge kommandot RUN C:0. Programmet fungerar nämligen så att en enhet med namnet C: definieras.

TIME.BAS

Program som sätter systemtid. Programmet gör CHAIN till programmet MENY, vilket ingår i kassett 2.

KASSETT NR 2

CASDISK.BAC

Program som kopierar filer från band till disk. CASDISK kopierar alla filtyper såväl text-filer som ABS- och BAC-filer. Du som använt CASTODSK för att kopiera från band till disk bör övergå till CASDISK för att slippa obehagliga överaskningar.

CASLIB.BAS

Programmet som går på grundutrustning ger en snygg utskrift av biblioteket i två spalter. CASLIB läser blockvis och kan alltså klara alla filer.

MASKEN.BAS

Programmet som nått herostratisk ryktbarhet, kanske inte främst på grund av märkvärdiga egenskaper, utan fastmer på grund av den berämda 'stölden'. Masken är dock värd en bekantskap, så prova och hjälp masken Max att hitta något att äta! Av speciellt intresse är den repetitionsfunktion som finns i programmet.

LABYRINT.BAS

Ett tredimensionellt labyrintspel i två dimensioner.

GOMOKU.BAS

Ett luffarschackprogram. Vinner alltid (?).

INVASION.BAS

Spelprogram, luftangrepp, med möjlighet att slå tillbaka.

HINDLOPP.BAS

Cursorn försöker ta sig över skärmen och möter uppdykande hinder.

DECROM.BAS

Programmet omvandlar decimala tal till romerska. Det är enkelt att använda och allmänbildande.

BLINKA.BAS

Mer musik i ABC80, liknar SLOOP i programmeringshänseende.

GUBBENO.ABAS

Ännu mer musik i ABC80!

NYMEMDMP.BAS

Disassembler för ABC80. Programmet som har en mängd funktioner är mycket användbart för den som vill lära mer om maskinspråk.

EXTDEL.BAS

Program som gör delete eller extract av rader i ett annat program. Programmet finns beskrivet i nr 1 1982 av ABC-bladet. Där finns också källkoden till de maskinspråksrutiner som används i programmet.

HUVUD.BAS

Detta program ställer en rad frågor som du bör besvara varje gång du skrivit ett program. Resultatet blir ett programhuvud som du kan få ut på disk eller kassett.

MENY.BAS

Dynamiskt meny-program tyvärr bekajat med något fel som får datorn att låsa sig. Programmet är ett exempel på vild programmering med sådana läckerheter som inhopp i loopar, flera subrutiner efter varandra utan föregående END osv. Vi har dock inget att invända mot programtiden som sådan och ser den gärna fullföljd på ett strukturerat vis om någon känner sig hägad.

TIME.BAS

Sätter systemtid och gör CHAIN till MENY.

LÄSDISK.BAS

Blockvis läsning på disk, du väljer drive, spår och sektor. Ett bra program för dig som vill veta hur lagringen på flexskiva går till. Programmet kan dessutom knappast skada dina skivor så var ej orolig för att prova.

FILOMV.BAS

Minne 16 Kbytes för flexskiva Programmet omvandlar en BAS-fil till BAC-fil eller tvärt om. Typiskt bra om man har lagrat ett program med list, men råkat få en rad längre än 117 tecken! Du som är osäker på vari skillnaden mellan BAS och BAC filer består bör dock först inhämta denna visdom.

DISKST.BAS

Programmet som egentligen borde ha hetat DISKSTAT

REGISTER.BAS

Sekvensiellt registerprogram med utskriftsmöjlighet.

DIRFIX.BAS

Räddar libben vid haveri.

OMVANDLA.BAS

Programmet kan köras på grundutrustning och är ett omvandlingsprogram mellan olika talsystem. Omvandling kan ske mellan binära, oktala, decimala och hexadecimala tal.

TNSET.BAS

Program av (c) Göran Sundqvist tel.0756/-30310 Minne 16 Kbytes och skrivare Testutskrift av alla tecken på en skrivare, med uppställning i ASCII-tabell.

SNYGGBAS.BAS

SNYGGBAS ÄR ett program för vettigare utskrift av BASIC-programmen än vad ABC-80's "operativsystem" bjuder på! Filer, som skall skrivas ut, lagras i ASCII-format med LIST-kommandot.

COPYLIB.BAC

Detta program ersätter ABS-programmet med samma namn och ger dessutom möjlighet att kopiera blockvis från disk till kassett. Programmet är viktigt eftersom det är en förutsättning för att vi ska kunna distribuera alla slags program på kassett. Tillsammans med CASDISK blir det ett paket som gör alla ABC80 med bandspelare kompatibla.

COPY.BAC

Program för kopiering av filer på disk. Programmet ersätter COPYLIB då enstaka filer ska kopieras mellan DR1: och DR0: eller när en fil ska kopieras med nytt namn på samma skiva. Observera att programmet ej innehåller något skydd mot att du 'bombas' en fil genom att ge samma namn åt utfilen som en redan existerade fil.

LIB.ABS

LIB-program i assembler att köras under det speciella operativsystemet för ABS-program, som du laddar in med BYE. Därefter kan du ge kommandot LIB.

DOSGEN.ABS

Detta dosgen-program bör användas av dig som har enkel densitet annars kan du få problem vid ombyggnad till dubbel densitet. Det kan exempelvis visa sig omöjligt att läsa gamla disketter som formaterats med det gamla dosgen.

TV.TXT

Innehåller kort information om TV-editorn med förteckning över vanliga kommandon.

TV.BAC

Detta program startar TV-editorn i floppy-versionen för 40 kol skärm. TV.BAC läser in TVSUBR och gör CHAIN till TVMAIN, alla tre filerna måste alltså finnas på disken.

TVMAIN.BAC

Huvudprogram i den s k TV-editorn och finns i flera versioner, varav den som ligger på ABC-kassett nr 2 är den första. TVMAIN måste sparas i BAC-format eftersom programmet innehåller en funktion på rad 10 som ej kan återläsas om programmet sparas med LIST.

TVSUBR.ABS

Masinspråksrutiner i ABS-format för TV-editorn, vilket innebär kompakt lagring av maskinspråk jämfört med DATA-satser och i all synnerhet POKE-satser. TVSUBR måste kopieras till skiva med CASDISK, det går ej med LOAD eller CASTODSK.

Kassett tre

CASDISK.BAS

Samma program som på kassett nr 2 men nu i BAS-format.

MUSIK.BAS

Orgel av tangentbordet, tyvärr med ganska svagt ljud. Det bättre sätt att få musik ut ur ABC80 vi hoppas återkomma på en senare kassett.

LANTERNA.BAS

Här får du lära dig vilka ljus olika farkoster ska föra i mörker till havs. Programmet prövar dina kunskaper genom att visa olika farkosters ljus som de syns i mörker. Din uppgift blir att gissa vilket slags farkost det är fråga om. Eftersom du inte vet vilken kurs de fartyg du möter har kan det vara nog så svårt. Du har dock hjälp av en programfunktion, du kan rotera de farkoster du ser. Tyvärr bara i steg om en grad. Detta kan du dock ändra på genom att ändra värdet på variabel z

UPÅT.BAS

Sjökrig, ubåt med torped, vanliga båtar med sjunkbomber.

BRAMUSIK.BAS

Musik på det vanliga viset som du känner igen från tidigare kassetter.

ELMÄTARE.BAC

Ett program!

CHECK.BAS

Med det här programmet kan du testa din ABC80's funktioner. Programmet har följande möjligheter; tangentbordstest, bildskärmstest, rom-test, kassett-test och printer-test.

ASSIGN.BAS

Detta program visar enhets-tabellen och gör det också möjligt att ändra enhetsnamn.

CASPROMP.BAC

Registerprogram för kassett som kan användas för t ex medlemsregister i mindre föreningar, men naturligtvis också för personligt bruk som att hålla ordning på grammfonoskivor och dyligt.

CASBLOCK.REM

Text som beskriver programmet CASBLOCK.

CASBLOCK.BAS

Program som kan rädda en fil med för långa rader ERR 20. Handhavande är ganska omständligt men lärorikt, vi hoppas att det manar till efterföljd till CAS-fantomernas båtnad.

MORSE.BAS

Program för den som vill träna telegrafi. Programmet påverkar V24-porten och kräver viss extra hårdvara t ex en summer ansluten till V24-porten. Närmare beskrivning finns på annat ställe i tidningen.

ABCV24.BAC

Här får du en printer-terminal rutin som gör kommunikation via V24-porten. Programmet simulerar en 80 kol skärm och kommer med en ny version av ABCTRANS att kunna köras på ABC80 med 80 kol som 80 kol terminal möjlig.

CASSEND.BAS

Kassettprogram för filöverföring till exempelvis klubbens monitor.

TVEDIT.REM

Textfil med information om TV-editorn, återfinns utökad på sid 6-7 i ABC-Bladet nr 1 1982.

TVCAS.BAC

Program för start av TV-editor på kassett, TVCAS gör CHAIN till TVSUB och TVMAIN i nu nämnd ordning.

TVSUB.BAC

Motsvarar TVSUBR.ABS i floppy-versionen av TV-editorn.

TV80.BAC

Motsvarar TV.BAC men fungerar på 80 kol skärm i floppy-versionen.

TVMAIN2.BAC

Utvecklad version av TVMAIN på kassett nr 2. Denna version kräver mer minne än den första versionen och bör ej användas av den som bara har 16 k. En nyhet är möjligheten att förflytta textstycken, som införts på denna version.

RADJUST.BAS

Program för justering av radlängd i en textfil inför utskrift. Medger utskrift med rak högermarginal på icke-proportionell skrivare. Tillsammans med TV-editorn ger RADJUST medlemmarna ett kraftfullt ordbehandlingsprogram utan stora kostnader.

REGCASPR.BAS

Registerprogram för kassett.

DISASEM.REM

Textfil med information om disassembler-programmet som kan skrivas ut på printer eller läsas med TV-editorn. För den som inte vet vad assembler och maskinspråk är kan det hela förefalla mycket mystiskt. Särskilt sedan man startat programmet! Tyvärr kan vi inte här ge en fullständig kurs i maskinspråk, utan måste återkomma i ämnet på annan plats. Var dock inte alltför orolig det finns proffs som inte kan maskinspråk.

DISASEM.BAS

OPLIST.BAS

Detta program skriver ut en lista på opcodeer och mnemoniks.

OPCODE1.TXT

Denna fil innehåller de 253 första opcodeerna med mnemo-koder. Om du är intresserad av dem kan du läsa filen med VISA som fanns på kassett ett.

OPCODE2.TXT

Denna fil innehåller mnemoiks för de opcodeer som kräver mer än en byte



SVENNE

Många har undrat vem Svenne är. Man har gissat hit och dit. En del har gissat på 495, men det är fel. Det är inte heller 1352 och naturligtvis inte 681! För övrigt kan jag avslöja att Svenne inte alls heter Svenne. Han vill bara vara anonym.

Nå till saken. Härom kvällen när jag stod med tandborsten i högsta hugg så ringde Svenne. Jag visste det innan jag svarade. Telefonen låter på ett alldeles speciellt sätt när Svenne ringer. Naturligtvis hade Svenne problem, jätteproblem, och bara jag kunde lösa dem, påstod han. Jag vet av erfarenhet att när Svenne börjar med att smickra så är det besvärliga frågor att vänta. Jag la alltså ifrån mig tandborsten och satte mig.

Svenne:

Jo du förstår jag har läst om maskinspråk och undrar hur man gör egentligen.

K-Å:

Jasså inget annat!

Svenne:

Jag har hittat ett maskinspråkprogram i en Amerikansk tidskrift. Det är för Z-80, så det ska väl gå på ABC80, eller hur? ABC80 har ju Z-80 processor.

K-Å:

Du har rätt ABC80 har Z-80 processor!

Svenne:

Just det, och nu ska du tala om för mig hur jag knappar in det här programmet!

K-Å:

Vad är det för ett program och hur ser det ut?

Svenne:

Det är en Pascal-kompilator. Det ser ganska lattjo ut! Men det är ju maskinspråk! Nu vill jag veta hur man gör jag får bara ERR 11.

K-Å:

Vill du läsa upp något ur programmet.

Svenne:

Hjärtans gärna! Här till exempel LD HLK, EF10H! När jag skriver det får jag ERR 11, vad gör jag för fel?

K-Å:

Du gör flera fel. För det första ska du inte skriva någonting av det där programmet för det förstår inte ABC80. Anledningen är att programmet är ett s k källprogram. Det är endast en källa till ett maskinspråk-program. Det där som ser så lattjo ut måste översättas för att bli maskinspråk

Svenne:

Jamen det måste väl gå att få in det på något sätt.

K-Å:

Både ja och nej. Den där Pascal-kompilatorn kommer inte att fungera på ABC80 om den inte är skriven för ABC80.

Svenne:

Ja men det är ju samma processor! Ett maskinspråkprogram för Z-80 ska väl fungera på alla maskiner som har Z-80!

– ett samtal

K-Å:

Inte alls, processorn är viktig naturligtvis, men för att ett program ska fungera på två olika maskiner måste en rad andra villkor vara uppfyllda. Ett program måste kunna kommunicera med tangentbord, in- utportar, interface och bildminne, och de sakerna ligger på olika ställen i olika maskiner.

Svenne:

Det där förstår jag inte. Är inte tangentbordet anslutet på samma sätt i olika datorer?

K-Å:

Många datorer har inte ens ett tangentbord, i stället ansluter man en terminal till datorn via ett interface. I sådana datorer finns inte heller något bildminne utan utmatning sker via terminalen.

Svenne:

Så den här Pascal kompilatorn kommer inte att fungera på ABC80?

K-Å:

Det går inte att svara på så här direkt. Det kan tänkas att man kan ändra i programmet, men det innebär sannolikt mycket arbete.

Men låt mig nu tala om någonting annat, nämligen maskinspråksprogrammering i allmänhet. Gissar jag fel om jag påstår att du nog inte har skrivit någonting i maskinspråk?

Svenne:

Nej det har du väl rätt i, men BASIC har jag ju lärt mig själv. OK du hjälpte mig lite i början. Ska det vara så mycket svårare med maskinspråk?

K-Å:

Kanske inte svårare men annorlunda. Det är några grundläggande saker man måste lära, som skiljer maskinspråksprogrammering från BASIC-dito.

Till att börja med måste du skilja mellan maskinkod och källkod. Det du såg i den där tidningen är källkod.

Svenne:

Jag förstår inte varför allt måste vara så krångligt, och hur ser egentligen maskinspråk ut?

K-Å:

Man skulle kunna säga att du hoppar över tio års datorhistoria när du lär dig BASIC, och tio år är mycket i datorsammanhang. I datorernas barndom fanns endast maskinspråk. Programmeringsspråk som ALGOL, FORTRAN och BASIC infördes för att underlätta programmeringen av datorer.

Svenne:

Men det finns väl olika maskinspråk, beroende på processor?

K-Å:

Men det finns goda skäl för att se maskinspråken som en grupp och de andra jag nämt som en annan grupp av språk. Man talar om låg- respektive högnivåspråk. Du har lärt BASIC - ett högnivåspråk - och försöker applicera de principer, som gäller för högnivåspråk när du ska lära ett språk på låg nivå. För att du ska kunna lära vad maskinspråk är måste du lära principerna för dessa lågnivåspråk.

Svenne:

OK! Förklara då vad det är för vits med källkod som inte kan matas in.

K-Å:

Om du vill titta på ett maskinspråksprogram så finns ett inbyggt i ABC80, BASIC-tolken.

Med följande programsnutt kan du titta på vad tolken innehåller.

```
10 ; "Startadress"
20 INPUT A
30 ; "Minnesinnehåll"+SPACE$(6)+
  "Minnesadress"
40 ; TAB(6) PEEK(A) TAB(23) A
50 A=A+1
60 GET B$
70 GOTO 40
```

Programmet ger innehållet i en minnescell varje gång du trycker ned en tangent. Du kommer då att finna att minnescellerna innehåller tal inom området 0-255.

Svenne:

Ja jag har sett det där redan, men jag förstår inte hur det där kan vara ett program. Jag menar, hur sjutton kan en massa nuffror göra någonting?

K-Å:

Därför att en del av nuffrorna i minnescellerna är instruktioner. Man brukar tala om operationskoder. För nybörjare på maskinspråk är det enklast att arbeta med decimala operationskoder eftersom PEEK och POKE fungerar decimalt.

Svenne:

Jag är inte säker på att jag förstår. Kan du inte ge exempel på någon maskininstruktion som jag kan skriva in och köra?

K-Å:

En viktig instruktion är RET eller return. Operationskoden för RET är 201 decimalt. Något förenklat kan man säga att den instruktionen ger ett återhopp till adressen efter senaste CALL-instruktion. Med CALL avses de olika CALL-instruktionerna i Z-80 koden.

Svenne:

Finns det inte en CALL-instruktion i BASIC också? Vad är skillnaden.

K-Å:

Det finns massor av skillnader. En CALL-sats i BASIC skrivs som vilken annan BASIC-sats som helst. CALL i BASIC används när man vill anropa en maskinspråksrutin. En maskinspråks-rutin kan anropas med följande programrad:

```
10 A=CALL(n)
där n är en adress i minnet
```

Svenne:

Nu är vi där igen! Hur får man in sin maskinspråks-rutin?

K-Å:

Ett sätt är att använda POKE-satsen. Med POKE kan du stoppa in ett tal i en minnescell. Med POKE 65408,201 så stoppar du in talet 201 i minnescellen 65408. När du gjort det så har du 'fått in' din första maskinspråks-rutin i din ABC80. Det som nu står i cell 65408 är en operationskod för en maskinspråksinstruktion, som heter RET. Instruktionen RET är en order om återhopp till exempelvis BASIC.

Svenne:

Jag förstår inte hur ett program kan bestå av bara 201, förklara dig!

K-Å:

Om du skriver POKE 65408,201 och därefter A=CALL(65408) så kommer 'ABC80' och blinkande kursor upp på skärmen. Det vill säga att du har kommit tillbaka till BASIC. Om du skriver något annat än 201 och gör CALL enligt ovan så går ABC80 vilse. Instruktionen RET används därför ofta som sista instruktion i maskinspråksrutiner som anropas av ett BASIC-program.

Svenne:

Jag vill lära mig fler instruktioner, kan du inte ge några exempel!

K-Å:

En instruktion är NOP med operationskoden 0. Denna instruktion som heter no operation, innebär att inget 'händer' när den utförs. Du kan nu skriva POKE 65408,0,201 och därefter A=CALL(65408). Du kommer blixtsnabbt tillbaka till BASIC. När du gör CALL till adress 65408 så anropar du instruktionen NOP, som du pokat in på den adressen. NOP utförs, vilket innebär en kort väntetid, men ingen förändring av minnesinnehåll och register förutom instruktionsadressregistret. Instruktionsadressregistret kallas också programräknare PC. Programräknaren får nu innehållet 65409 och pekar alltså på nästa instruktion (0 dvs NOP) som omedelbart hämtas och utförs. Därefter utförs nästa 201 dvs RET och du kommer tillbaka till BASIC.

Svenne:

OK! Nu har du berättat om två instruktioner 0 som heter NOP och inte gör något och 201 RET som man har i slutet på rutiner. Men det måste finnas fler instruktioner, jag gissar att det finns 200 till, nämligen 1-200! Finns det inte någon förteckning någonstans?

K-Å:

Det kommer en förteckning över de 253 första i nästa nummer av Bladet. Dessutom tror jag vi måste bryta här. Innan vi kan gå vidare måste jag nog förklara hur processorn fungerar i stora drag, något som jag också vill spara.

Kjell-Åke Johansson



PROGRAMMERAR-HJÄLP

DU som kan programmera ABC80 med IEC-buss och vill jobba några timmar i veckan.
Lämpligt för studerande.

Ring och prata med:
Guy Portnoff
KVARTSELEKTRONIK AB
08 - 37 25 30
BROMMA



BREV

Följande brev fick vi 5 maj 1982 med kommentaren att LUXOR ej svarat och att ABC-bladet fick publicera det

Bernhard Granholm
Ljunggatan 39 C
664 00 GRUMS

GRUMS 1982-03-28

Följande systematiska fel har upptäckts på ABC80. Om kod för DATA-rad finns i variabel-listan så kan datorn uppfatta denna som början på en DATA-rad och läsa följande tecken med en READ-sats.

Här är en beskrivning på det inträffade, alltså hur det kan fungera i ett program. I bifogat utdrag ur programmet finns på rad 220 en ONERRORSATS. Av bekvämlighetsskäl har jag satt loopen med READsats med fler omlopp än vad som finns variabler i DATA-raderna (utrymme för fler variabler förberett. När DATA-raderna är slut kommer ERROR 30 vilket tas om hand av ONERROR-satsen. Detta fungerade tills efter en ändring i programmet det i stället blev ERROR 10 LINE 230 (Texten får ej plats i strängen). Jag dimensionerade upp strängvariabeln och fick så tag i den. Efter CTRL C och manuella kommandon skrev jag den på diskett. Därefter jämförde jag med innehållet i arbetsminnet och hittade "variabeln" i variabellistan med början på adress 38792. Adressen före innehåller decimaltalet 147, som är koden för DATA-rad och i detta fall ingår 147 i pekare till adress 38803 (nästa variabel) ($38803 = 147 + 256 * 151$). Datorn är försedd med 16k extra arbetsminne.

Felet försvann när jag lade till en rad i programmet, t ex en REM-sats. Felet återkommer varje gång programmet har en längd så att adresspekaren innehåller talet 147.

För jämförelse sänder jag med ett utdrag ur arbetsminnet med en riktig DATA-rad (börjar t ex vid adress 38322 med textbörjan BORLÄNGE fr o m adress 38326).

```

2200 DATA 10.7
2205 DATA 92.2,96.7,98.95
2210 DATA 90.6,94.9,97.9
2220 DATA 88.9,93.7,95.7
2230 DATA 88,91.1,93.4
2240 DATA 91.4,96.2,99.7
2250 DATA 90.2,94.3,98.65
2260 DATA 89.2,92,97.55
2270 DATA 32078,31775,32272,32007,31745,32258,31787
2280 DATA MORRA,SVEG,SARNA,IDRE,GROVELSJON,DREVDAGEN,GORDALEN
2290 DATA BORLANGE,HOLJES,TRANSTRAND
2300 DATA RORBACKSNAS
2310 DATA ANGE,OSTERSUND
2320 REM * FILNAMN
2330 IF LEN(FR)>8% FR=LEFT$(FR,8%)
2340 FR=FR+'.DAT'
2350 RETURN
2360 REM * LASER FRAN FIL
2370 ONERRORGOTO 2440
2380 OPEN FR ASFILE %
2390 INPUT %X,NB
2400 FOR IX=1% TO %X
2410 INPUT %X,B(IX)
2420 GOSUB 2530
2430 NEXT IX
2440 CLOSE %
2450 RETURN
2460 REM *
2470 FOR IX=1% TO %X
2480 B(IX)=F(IX,IX)
2490 GOSUB 2530
2500 NEXT IX
2510 NB=B(IX)
2520 RETURN
2530 : CUR(IX+5%*IX+7%*NB)B(IX) : RETURN

```

Programlista i INTERNKOD RAD 2290

Del av program med READ-satsen RAD
230

NgigCz lzfIC0 |rCO A -DA P * ID# DMK b- I * || . _ @ |@J / | # = | → _ F ,

Variabeln skriven med METRIC 8500

151 94 153 103 0 5 0 25 0 3 0 242 67 157 151 160 0 102 155 9 0 246 67 167 151 6
156 25 0 0 0 245 87 177 151 170 158 25 0 2 0 244 68 187 151 222 158 12 0 5 0 24
6 68 197 151 31 159 12 0 0 0 240 77 204 151 6 112 0 0 130 241 73 222 151 14 0 2
46 128 210 151 173 128 7 0 1 0 241 74 238 151 0 0 232 128 226 151 195 128 3 0 1
0 248 70 6 152 113 160 120 0 5 4 10 0 10 105 130 210 151 153 129 25 0 1 0 242
86 16 152 80 0 206 162 1 0 241 86 22 152 0 0 241 67 38 152 0 0 202 130 210 151 1
57 130 0 0 1 0 81 82

Variabeln i ASCII-kod

38780	66
38781	90
38782	152
38783	212
38784	185
38785	58
38786	145
38787	13
38788	1
38789	248
38790	67
38791	147
38792	151
38793	94
38794	153
38795	103
38796	0
38797	5
38798	0
38799	25
38800	0
38801	3
38802	0
38803	242
38804	67
38805	157
38806	151
38807	20
38808	0
38809	102
38810	155
38811	9
38812	0
38813	246
38814	67
38815	167
38816	151
38817	122
38818	155
38819	25
38820	0
38821	6
38822	0
38823	245
38824	67
38825	177

147 folkas som börjar på DATA-rad

38803

Pekar dit (nästa variabel)

Variabellistan

Vid förfrågan i början
tember hade inte L.sv.
Samt "Jag har provkör
met också i en ABC80
minne. Det händer samm

claes s.

Q-Zentralen

har nu kommit i gång. Det är flera medlemmar som tagit fasta på erbjudandet i ABC-bladet 1982:1.5.

Genom Q-Zentralen får medlemmarna tillgång till telekonferenssystemet KOM som utvecklats av FOA. Vissa inlägg har tidigare återgivits i ABC-bladet. Det finns bl a en särskild konferens som heter **ABC-80 erfarenhetsutbyte**. Vidare finns **ABC-klubben Medlemsforum**, **ABC-novis Svenne** och **Forth Erfarenhetsutbyte**. Man behöver inte alla vara en erfaren ABC-användare för att ha nytta av Q-Zentralen. Det räcker att ha tillgång till modem. Det är inte svårare att köra på QZ än att köra på ABC-klubbens Monitor. Mötet ABC-novis Svenne är avsett för frågor av den typ som "Svenne" ställt i spalten **Samtal med Svenne** i ABC-bladet. Där du kan ställa frågor och snabbt få svar från andra användare. Gå med i andra telemöten och ta del av de diskussioner och den debatt som förs. Du kan öppna ett eget möte i ett ämne som intresserar dig särskilt.

Det är vissa villkor förknippade med utnyttjandet av Q-Zentralen. De framgår av den ansvarsförbindelse alla måste underteckna, där står bl a följande:

"Min verksamhet på Q-Zentralen är av hobbynatur, och har inget kommersiellt intresse. Kostnaderna för verksamheten kan därför inte betalas av något företag eller dras av i inkomstdeklarationen. Min verksamhet är inte heller avsedd att ge någon vinst. - - - Jag förbinder mig att betala in körkostnaderna utan anmaning till ABC-klubben på särskilt inbetalningskort, minst en gång/månad då körning pågått."

QZ tänker sig att Q-Zentralen skall kunna vara en central Monitor med programbank för alla smådatoranvändare. Avsikten är att ABC-klubben skall svara för att lägga upp och underhålla en central programbank för ABC-80. Vi kommer att successivt föra över "screenade" program till Q-Zentralen. QZ står för kostnaderna för datalagringen.

Det finns två olika konton eller kundnummer som medlemmarna får utnyttja, ett till lågtaxa öppet efter kl 20.00, där man endast kan köra KOM och få tillgång till programbanken och ett till något högre taxa där man tämligen fritt får utnyttja DEC-10. Vid den lägre taxan kostar det fn ca 20 kr/per timme att köra KOM. Att köra KOM till den högre taxan skulle kosta ca 50 kr/tim. KOM kostar relativt mycket i jämförelse med annan körning, därför blir den genomsnittliga kostnaden sannolikt lägre men det beror naturligtvis på vad man gör.

Tills vidare är det fråga om en försöksverksamhet, men styrelsen verkar för att verksamheten skall bli permanentad.

Vill du få tillgång till Q-Zentralen gör så här:

Betala in 50 kr i inträdesavgift på ABC-klubbens speciella postgirokonto för Q-Zentralkörningar, pg nr 43 51 74-8. Då får du dig tillsänt **manual**, **kassett med program och exempel på KOM-trafik**, formulär till ansvarsförbindelse samt några inbetalningskort. När du returnerat ansvarsförbindelsen undertecknad får du ut password.

Äventyra inte ditt medlemskap i ABC-klubben!

ABC-kasset nr 4 har sänts ut till alla som erlagt stadgad medlemsavgift för 1982. ABC-kasset 5 och 6 är i det närmaste klara att sändas ut. Nr 5 innehåller fortsättningen på FORTH och på kasset nr 6 kommer bl a LISP. Dessutom räknar vi med att under året få klar ABC-kasset nr 7 med bl a en kassetversion av FORTH.

De kommande numren av ABC-bladet kommer att innehålla artiklar om bl a textbehandling, FORTH, programmeringstips och vanliga nybörjarfel.

Missa inte alla intressanta nyheterna genom att inte betala medlemsavgiften. Enligt stadgarna skulle avgiften ha erlagts före utgången av april. Inbetalningskort med anmodan att betala fanns i ABC-bladet 1982:1 som kom ut i månadsskiftet april-maj. En påminnelse med nytt inbetalningskort fanns med i Mini-bladet 1982:1.5 som skickades ut i juli. De som betalt har fått sina medlemskort.

Medlemskort för 1982 har sänts ut mot postförskott till de inom Sverige som inte betalt sin avgift (bokförd på postgirot senast 1982-09-16). Därvid har 5 kr lagts på för att täcka merkostnaderna för postförskottsavgiften. De som bor i utlandet har fått en förnyad anmaning.

De som inte har betalt eller löst ut sina medlemskort när postens liggetid har utgått kommer att avföras ur klubbens medlemsregister vilket bl a innebär att detta ABC-blad blir den sista försändelse som erhålles från klubben.

Tänk på att anmäla event **ändring av adress och telefon**. Juniorer skall uppges sitt personnummer. De uppgifter som vi har i vårt register kommer att ligga till grund för den medlemsmatrikel vi i likhet med tidigare år kommer att ge ut. Sista dag för ev komplettering av uppgifter till matrikeln är 1 nov.

Styrelsen

Nytt om Monitorn

På mångas begäran har kommandot **FILELIB** lagts in på Monitorn. Genom att ge detta kommando får du en LIB-lista på fil som du i lugn och ro kan studera hemma. LIB-listan ligger inte lagrad som fil i Monitorn utan den "tillverkas" när du ger kommandot och är därför alltid aktuell. Det spelar ingen roll vad du svarar på frågan om filnamn hos monitorn, huvudsaken är att du anger det filnamn under vilket du vill lagra listan hos dig. Du kan välja antal kolumner du vill ha i listan beroende på printer- eller skärmbredd.

Numera går det bara att logga in två gånger i direkt följd sedan måste du släppa in någon annan.

Under våren och försommaren har vi haft ett oregelbundet uppdykande fel, förmodligen hårdvarufel, som har varit mycket svårt att diagnosticera. Därför har Monitorn tidvis varit ur funktion.

En utförlig artikel om Monitorn fanns i ABC-bladet 1981:4/5.

Mötesverksamheten i Stockholm

Programmet för höstens mötesverksamhet i den nybildade Stockholmsavdelningens regi är nu i det närmaste spikat.

När du läser detta har medlemmarna i ABC-Stockholm redan fått personlig kallelse till det första av höstens möten, onsdagen den 6 oktober i **VIDÄNGSSALEN**.

Följande möten blir preliminärt:

Onsdagen den 27 oktober kl 19.00 i **VIDÄNGSSALEN** kommer **Bob Johnsen** och berättar om den **Fig-FORTH** som han gjort för klubben till ABC-80. Missa inte chansen att under Bobs sakkuniga ledning bekanta dig med det nya programspråket FORTH som du nu fått på ABC-kasset.

Onsdagen den 17 november kl 19.00 i **VIDÄNGSSALEN** berättar Gunnar Tidner om **textbehandling** samt informerar kort om QZ.

Onsdagen den 8 december i **VIDÄNGSSALEN** blir nästa möte. Skall vi gissa att kvällens tema kommer att röra klubbmedlemmarnas erfarenheter från användning av QZ.

Personlig kallelse utsändes till alla som är medlemmar i ABC-Stockholm. Eventuella ändringar i programmet meddelas i kallelsen och kommer dessutom att läggas in på **telefonsvararen** på 08-80 15 22, på Monitorn samt på Q-Zentralens KOM-möte "ABC-klubben medlemsforum" som är öppet för bl a annonsering av möten av detta slag.

Ingen inträdesavgift för medlemmarna i ABC-Stockholm, övriga 10 kr.

VIDÄNGSSALEN är belägen vid T-bana ALVIK ner för trapporna mitt emot bussarna, i **ALVIKS KOMMUNALHUS** VIDÄNGSVÄGEN 1.

Kontraktproblem

Det här brevet innehåller en beskrivning över hur ABC-klubben "drabbat" mig, en beskrivning av kommunikationsmissar, samt avslutas med några önskemål.

Jag hade, när jag genom mitt arbete kom i kontakt med ABC 80, hört talas om ABC-klubben. Det var svårt, men inte omöjligt att få tag på adress och telefonnummer. Det tog några dagar. Det var helt omöjligt att få något begrepp om vad klubben gjorde och gjort, att få någon sorts "varudeklaration".

Med mindre vetskap om klubben, än jag önskat betalade jag min medlemsavgift i slutet av januari 1982. Att få tag på postgironumret, medlemsavgiftens storlek och tillvägagångssättet för att bli medlem tog mig ytterligare en dag.

Det tog lång tid innan jag hörde något mer från klubben. Så småningom dök ett tillfälligt medlemskort upp. Det berättades om **Monitorn**, där man kunde hämta massor med fina program. Hur man kom i kontakt med den, det framgick inte.

Flera gånger hade jag försökt ringa till klubbens telefonnummer utan resultat. Så en dag stod mig lyckan bi: Jag fick tala med **Joe Johnsson**. Joe berättade vad **monitorn** var, han berättade att utgivningen av klubbtidningen försenats och att det skulle komma ett riktigt medlemskort. Han berättade också att medlemsavgiften höjts, men att jag inte behövde betala mer då jag betalat före årsmötet. Dessutom berättade han om ABC-rapport nr 1. Den har jag haft glädje av. Joe lämnade sitt telefonnummer. "Ring om du har bekymmer, det är OK!".

Så småningom hade jag ett 300 Baud modem, och nyfikenheten på **monitorn** växte igen. Det visade sig åter vara omöjligt att få något grepp på hur det hela funkade. Jag tyckte nog att ABC-klubben var en tveksam företeelse. Svårt att få kontakt, svårt att få besked. I desperation ringde jag Joe igen. Denna ABC-klubbens räddande ängel!

Genom Joe fick jag tillgång till flera tidigare ABC-blad och också en del program. Nu skriver jag med tv-editorn, och jag har lyckats komma in i **monitorn**. Inte tack vare ABC-klubben, utan tack vare Joe!

I dagarna har jag fått ABC-blad 1.5/82 - fortfarande saknar jag dock nr 1/82. På mitt blad står det inte "Bet 82" på adresslappen, trots att jag betalade medlemsavgiften 1982-01-27 via postgiro. Det gör mig orolig för att jag skall behöva kämpa mig till mina medlemsförmåner, på samma sätt som jag kämpat mig till information hittills.

Jag har sett i ABC-bladet att man är fundersam över vikande och för litet medlemsantal. För mig är det ett under att klubben har så många medlemmar den har, med tanke på hur besvärligt det är att ens få upp ett "doftspår" av klubben!

Hade jag inte haft turen att träffa på Joe skulle min uppfattning om ABC-klubben vara klar: INTE VÄRT PENGARNA! Genom Joe har jag fått tillfälle att se vad klubbens medlemmar åstadkommit, och det är imponerande.

Uppfattningen om ABC-klubben är alltså utlämnad till om man har turen och orken att ringa tills man ev har turen att hitta sin Joe.

Förändringsförslag

1. I marknadsföringen av föreningen måste framgå **varför** jag skall vara med i föreningen. Det skulle räcka med ett litet urval av de godsaker jag hittat i ABC-bladen.

Det måste också framgå **hur** man bär sig åt för att bli medlem. Klubbens telefonsvarare får inte bara säga "Bandet är slut, ring senare - hur stänger jag av den hä?" (så lät det i somras). Helst borde det finnas en hänvisning till ett telefonnummer, där man får svar och besked. I varje fall borde det framgå hur man blir medlem!

2. Varje nybliven medlem borde förse med ett "initialmaterial", där bl a **monitorn** presenteras. Materialet borde också kunna innehålla ett programurval på kassett och en presentation av vad som tidigare givits ut på band och papper, så intresserade ges möjlighet att beställa det.

Jag tycker det skulle vara tragiskt om klubben förlorar eller snarare missar nya medlemmar p g a sådana här elementära taffligheter. En klubb som erbjudit så oerhört mycket, efter vad jag kunnat se i mina ex av gamla ABC-blad!

Med vänlig hälsning

Lars Frej <2260> ALNÖ

Svar till Lars Frej m fl

Styrelsen beklagar att servicen till nya medlemmar ibland fungerat dåligt. Klubben har bara två år på nacken och den stora (men givetvis glädjande) ökningen av antalet medlemmar har medfört att rutiner och hantering har förändrats. Det har bl a varit nödvändigt att ändra uppläggnings av klubbens medlemsregisterhållning mer än en gång.

ABC-klubben sänder under året ut ett ganska omfattande material av ABC-blad, ABC-kassetter och matrikel som masskorsband till alla medlemmar. Styrelsen har ansett att arbetet med att ta fram underlag till, producera och distribuera detta material måste komma i första hand.

Härtill kommer den direkta servicen med enskild försändning av medlemskort, tidningar, kassetter, rapporter, manualer mm som numera har blivit mycket omfattande. För att i någon mån underlätta arbetet gör vi i ordning ett paket med samlingsnummer av 1980-1981 års tidningar samt ABC-kassett nr 1 och 2 som säljes för 100 kr.

Klubben är i ett uppbyggnadsskede och projekt, ideer och vilja "viner runt knutarna".

Lars Frej beskriver hur han mötts av en telefonsvararen när han gång efter annan ringt till klubblokalen i Alvik. Det är helt riktigt. Klubblokalen är inte bemannad. Klubben har dock numer skaffat en telefonsvarare som kan ta emot meddelanden. Naturligtvis kan ingen telefonsvarare i världen ersätta ett bemannat klubbkansli. Lars har ändå haft en ovanligt stor tur som fått direktkontakt med en styrelsemedlem som råkat befinna sig i klubblokalen vid ett av hans telefonsamtal.

Det löpande arbetet med ABC-blad/kassetter, det utvecklingsarbete som sker samt den ovriga klubbadministrationen har hittills skötts av oss i styrelsen på vår fritid. Klubben har kanske snart nått en sådan omfattning att pengar måste tas fram för att bemanna kansliet i viss omfattning.

Det är bra att enskilda medlemmar hör av sig med synpunkter och förslag. Det är ett sätt att medverka till att förbättra klubbens verksamhet. Hör därför gärna av er! Det går bra att ringa, våra telefonnummer finns lättillgängliga på matrikelns baksida. Det går också bra att skriva helst i maskinläsbar form, gärna med TV-editorn på kassett. Du kan också göra "insändare" på **Monitorn** eller varför inte inlägg i **KOM/Q-Zentralen**.

Vi tror att det vore en fördel om kontaktvägarna inte enbart gick medlemmar-styrelse utan även medlemmar-medlemmar. Många brev och telefonsamtal rör konkreta problem alltifrån mycket enkla frågor från noviser till de mest intrikata tekniska frågeställningar där vi inte alltid vet svaren.

När vi bedömer att en fråga vi fått har allmänt intresse väljer vi kanske att lämna ett svar i ABC-bladet. Räkna inte med att vi har resurser att ge personliga svar på sådana brevfrågor.

Vi tycker inte att du skall tveka att ta **direktkontakt med en annan ABC-medlem** i din närhet. Även om ni inte känner varandra tror vi inte att någon ABC-medlem skulle låta bli att ge en hjälpsam hand om det står i hans eller hennes förmåga. Skälet till att matrikeln även innehåller en medlemsförteckning i postnummerordning är bl a att underlätta denna typ av tvärkontakter.

Erfarenheterna från **Monitorn** visar att många medlemmar utnyttjar denna möjlighet att ställa frågor och få svar från andra medlemmar. Möjligheten att utbyta erfarenheter och föra diskussioner fram och tillbaka är ännu bättre på **KOM**.

Välkomstpaketet med initialmaterial är på väg, det beslöt styrelsen på försommaren.

Styrelsen

Hjälpa till med att 'screena' program?

Till årsmötet hade Styrelsen tagit fram ett förslag till en verksamhet som kom att kallas 'screening'. Förslaget publicerades i ABC-bladet 1/82 och innebär i korthet att nu när klubben börjar massdistribuera program på kassett måste programmets ursprung och skick kontrolleras. Detta i sin tur ger upphov till frågor om upphovsrätt. Det är viktigt att klubbens policy tar etiska hänsyn i dessa frågor. Detta har fått än större aktualitet genom öppnandet av Q-Zentralen för ABC-klubbens medlemmar.

Programmen

ABC-klubben får program på många olika vägar. Monitorn svarar för en stor del. Medlemmar skickar in en hel del program på kassett och som programlistningar. Vidare har klubben fått rätten att publicera några proffsprogram, som kommit på diskett. För alla dessa kanaler och former måste klubben skapa rutiner för att kunna ta hand om programmen och förmedla dem till klubbmedlemmarna på ett meningsfullt sätt.

Upphovsrätt och policy

På klubbens årsmöten har diskussioner förts om hur viktigt det är att klubben har ett strikt förhållningssätt till upphovsrätt av programvara och skydd mot orättmätigt intrång. Detta har fått styrelsen att i ABC-bladet ge uttryck för ett avståndstagande till programstöld.

Detta måste ses som ett överlevnadsvillkor för ABC-klubben. Om inte yrkesprogrammerare arbetar med ABC80 och proffsprogram tas fram, blir ABC80 snart "omodern". Detta är ju något som varje seriös medlem i klubben minst av allt önskar.

Däremot måste klubbens styrelse tolka åtgärden att skicka in program till klubben, antingen dess Monitor eller via kassett, som ett medgivande att klubben får rätt att disponera programmet ifråga.

Detta gäller när insändaren är upphovsman. Det är därför viktigt att klubben håller reda på varifrån de olika programmen kommer.

Brevledes torde detta inte vara så svårt. Beträffande Monitorn tillförs varje insänd fil ett identifikationsmärke, d v s medlemsnummret. Detta betyder också att ansvaret för att en fil ligger på Monitorn är insändarens.

Det kan dock inte uteslutas möjligheten att Monitorn får mottaga program eller filer som ej är helt 'fria'. Dessa kommer av förklarliga skäl med i råddumpen från Monitorn och behandlas i screening-gruppen. Programredaktören beslutar sedan om distribution överhuvudtaget skall ske och så fall i vilken form.

Klubben måste därför skaffa sig sådana försäkringar från projektdeltagarna att klubben denna väg ej kan pådyvlas att ha gjort intrång i copyrightbestämmelser och upphovsrätt.

Arkivfrågan

Den stora tillströmningen av program gör det också nödvändigt att se till att klubbens programarkiv och rutiner i samband med detta blir överskådliga och säkra. Utvecklas inte bra rutiner växer snart mängden program ut över det överblickbara och enbart uppgiftens omfattning kan skrämja hugade klubbmedlemmar att hjälpa till.

Deltagande och ansvar

En försöksverksamhet har pågått i Stockholm under vårterminen 1982 med regelbundna samlingar en gång i månaden. Vid dessa träffar har en avrapportering och en diskussion om klubbens policy och etik skett samt vad olika ställningstaganden härvidlag kunde föra med sig. Emellan dessa samlingar har arbetsmöten ordnats.

Av vad som tidigare sagt, framgår att klubben måste kräva en försäkran att man förstått projektets etiska linje och att man lojalt kommer att följa denna med de program som man kommer i befattning med.

Deltagare får ta del av någon annans bearbetning för att konfirmera rimligheten i denna redovisning och i samband med detta kontrastsignera redovisningen för klubbens räkning.

Du, som vill ingå i screenings-gruppen, kontakta klubben så sänder vi ett PM med instruktioner. Skicka gärna vykort och märk det tydligt med 'Screenings-gruppen' och ange (läsligt) Ditt medlemsnummer och adress. Det kan ju också vara bra att Du uppger Ditt telefonnummer så nås Du lättare om det skulle vara några frågor.

Styrelsen

Två nya ABC-kort

EPROG

Du programmerar
de flesta EPROM
- direkt från BASIC
— DATA —

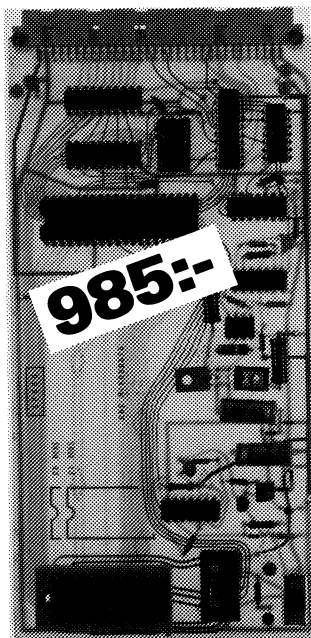
Tar bl.a.:

- * 8K 2758
- * 16K 2716, 2516
2724, 27C16
- * 32K 2732, 2532
- * 64K 2764, 2564,
68764
- * NOLLKRAFTSOCKEL
- * TYPMODUL ökar
säkerheten

TYPMODUL för 2716
+ ABC80-program på
kassett medföljer
Ytterligare typmodul
+ programkass.: 65:-

Kablage kort-tangentbord 175:-

Priser utan moms.



Ring eller skriv till:

PROTO

Bygg ditt eget
I/O-kort eller labba
Du styr från BASIC

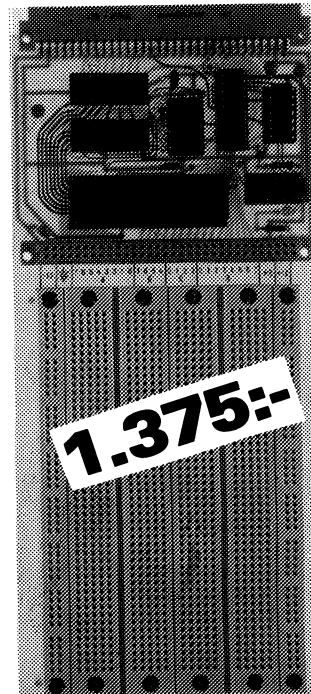
— DATA —

I/O-SOCKELN:

- * 24 I/O-linjer
2 terminaler/linje
4 mA per utgång
- * + 5 + 12 -12 Jord
4 term. vardera

LABBORDET:

- * 192 5-terminaler
- * 4 40-terminaler
- * Kortplatsen kan
lätt ändras
med DIPSWITCH
- * DOKUMENTATION
följer båda korten



ABC-Mikrodata
Forsakersg. 7, 431 33 Mölndal
Tel: 031-27 85 53

- * Förlängda
Europakort
- * ABC80-bussen



- ett grafikprogram med ljud för nybörjare
av Ferdinand Mican

När jag såg programsnutten "Pausfågel" så frågade jag mig varför ingen hade illustrerat ljudet.

Två flugor i en smäll eller två käpphastar i glad förening, nämligen programmering och fågelskådning. Nej, förresten var det många fler flugor. Samtidigt kunde jag ju lära mig lite om grafik på min mikrodator.

Att döva mitt samvete som ständigt säger att jag inte borde hänga bakom tangentbordet halva natten, det lyckades jag däremot inte med. Även om jag försökte efterlikna en av faunans trevligaste varelser.

Jag började med att skaffa mig kopior av Videografik-kartan, som finns i bruksanvisningen för ABC-80. Sedan ritade jag in konturerna på min fågel, som hör till familjen Sturnidae. Starar kan nämligen härma alla ljud, t.o.m. högtalarens gnissel som lätt kan förväxlas med en fönsterputsares arbetssång. Nästa steg var ifyllandet av samtliga punkter som hade drabbats av min blyertsenna. Nu kunde jag se hur bilden skulle se ut på rutan.

Att snabbt komma igång med knappandet, det är freakens melodi. (Freak uttalas frik. Det är engelska, förstås, och beteckningen för datorhobbyister. Betyder egentligen hugskott eller "originell människa".) Proffsens JSP-strukturerer behärskar jag inte så jag nöjer mig med att repetera avsnittet om grafik i "ABC om BASIC". Om Du undrar vad "JSP" är så kan jag tala om att det är en programspråks-oberoende konstruktionsmetod och förkortningen för Jackson structured programming.

I ABC-boken står det i alla fall att man först sätter hela skärmen i "grafisk mode". Det gör jag på raderna 110 och 120. Sedan försöker jag rita med en teknik som tilldelar ASCII-tecknen ett bestämt mönster om skärmen är i grafisk mode.

Snart upptäcker jag att det är ganska mödosamt att omvandla alla prickar till bokstäver och konstaterar att denna metod bäst ägnar sig för stora teckningar där en stor del av skärmen är täckt av "linjer". Nej, då är det bättre med punkter i ett koordinatsystem. Det är klart att vill man ha hela skärmen vitt så skriver man inte 5.616 gånger SETDOT(R,K) om man kan omvandla straffet i böter. På hela skärmen finns det nämligen så många grafiska punkter om man räknar ifrån vertikal-rad 0 där omvandlingen till grafisk mode osynligt ligger. R och K är siffrorna i det koordinatsystem som gör att man kan sätta alla punkter på bildskärmen. Som Du kan se på bild 1a så finns det 78 horisontella lägen och 72 vertikala. De första två lägena är nämligen reserverade för att kunna sätta skärmen i grafiskt mode.

Pausbird

Nu kan jag väl vara nöjd? Men det är med program som med en trädgård - den blir aldrig färdig. Det är så mycket man kan göra annorlunda och bättre. Programmet tar onödigt mycket minne. Genom att konsekvent använda heltal kan man spara 4 byte på varje siffra. Näbbens rörelse i ljuddelen har dragit ner hastigheten och jag måste skruva upp kråkan igen så att det blir lite fart på kvittret. Sedan kan man väl lägga in klockan i bilden och

Nej, nu får det räcka för den här gången. Nästa bild ritas jag med en helt annan teknik. T.ex. med programmet RITA som jag som medlem i ABC-klubben har fått. I detta program kan man med olika tangenter styra markören över bildrutan och på detta sätt rita en bild. Kanske morgondagens konstnärer slänger pensel och staffliet och köper sig ett äpple och en fröjdepinne.

Hjälp, det tar tid det här. Jag försöker snabba upp ritandet med hjälp av ED-kommandot. Jag får förstås ändra radnumret och siffrorna om jag vill få glädje av den editerade raden. Nu skulle man behöva en mänsklig medhjälpare som kan läsa upp siffrorna.

Ibland får man göra RUN för att kolla att man har knappat in rätt siffra. Samtidigt är det en liten belöning för mödan när man ser hur bilden växer fram.

Efter några timmar är jag då färdig med denna del av programmet och kan fortsätta med ljudet. I enlighet med god programmeringssed så skriver jag REM för ljudet så att jag inte kan glömma vad det är för konstiga saker som jag har skrivit. Sedan skriver jag looparna som bestämmer ljudet med OUT 6,5 och pauserna med OUT 6,0. Nu provar jag lite olika hastigheter för att lista ut vilket kvitter som blir bäst. Eftersom allt låter lika bedrägligt så skriver jag dit en siffra och bestämmer mig för att vara nöjd. Jag gör RUN igen. Pippin kommer fram och kvitrar och egentligen borde jag vara nöjd men jag tycker att det saknas något. Pricken över iet! Vad är detta? Den rör ju inte på sig, den är död den. Det är klart att den ska röra på näbben och flämta och flaxa. Men var någonstans i programmet lägger jag in denna rörelse så att den rör sig med samma hastighet som ljudet. I ljudets loop förstås. Där lägger jag in ett annat läge för näbben och nu tänder sig näbben med SETDOT och CLRDOT i överläge och undre läge. Flämtandet med bröstet får jag på veterinärens inrådan ta bort - det ser sjukligt ut.

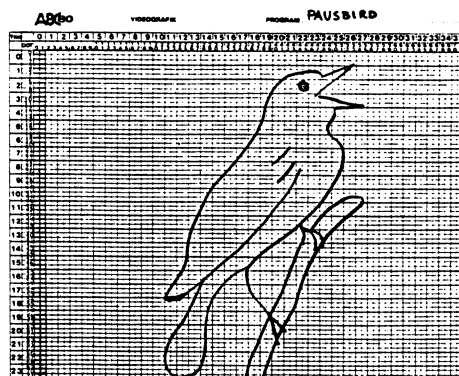


Bild 1b: Här är fågeln inritad.

Exempel på olika sätt att adressera bildskärmen
CURSOR, SETDOT direkt och med READ DATA SETDOT.

```
10 REM SAVE PAUSEBIRD.GRF
20 REM
30 REM *****
40 REM * PAUSFÅGEL FÖR ABC-80 *
50 REM * AV FERDINAND MICAN *
60 REM * TEL. 0510-623 36 *
70 REM * REV. 020326-1 *
80 REM *****
90 REM
100 ; INP(3);CHRS(12%)
110 FOR I%=0% TO 23%
120 ; CUR(I%,0%);CHRS(151%);
130 NEXT I%
140 ; CUR(1%,21%);"p,l8f"
150 ; CUR(2%,20%);"6"
160 ; CUR(2%,22%);"8,i"
170 ; CUR(3%,19%);"-"
180 SETDOT 9%,47% : SETDOT 9%,48% : SETDOT 9%,49%
185 SETDOT 9%,50% : SETDOT 9%,51%

660 READ R,K : SETDOT R,K
670 DATA 55,24,55,26,55,30,55,38,55,40,55,44
680 DATA 56,26,56,29,56,30,56,38,56,39,56,40,56,44
690 DATA 57,25,57,29,57,39,57,43
700 DATA 58,25,58,29,58,39,58,40,58,43
710 DATA 59,25,59,29,59,39,59,40,59,41,59,42
720 DATA 60,23,60,23,60,39,60,42
730 DATA 61,24,61,29,61,39,61,40,61,42
740 DATA 62,24,62,28,62,29,62,38,62,40,62,41
750 DATA 63,23,63,28,63,29,63,38,63,40,63,41
760 DATA 64,23,64,28,64,29,64,37,64,38,64,40,64

860 REM *****
870 REM * KVITTER OCH RÖRELSE *
880 FOR K=100*RUN TO 100*RUN
890 OUT 6,5
900 REM
910 REM ÖVERNÄBBEN-ÖVRE LÄGE TAS BORT
920 CLRDOT 4,49 : CLRDOT 4,50 : CLRDOT 3,51
930 CLRDOT 3,52 : CLRDOT 4,52 : CLRDOT 5,51
940 CLRDOT 6,50 : CLRDOT 7,49 : CLRDOT 7,48
950 CLRDOT 5,50 : CLRDOT 3,53 : CLRDOT 3,51
960 REM
970 REM ÖVERNÄBBEN-NEDRE LÄGE SÄTTS
980 SETDOT 5,49 : SETDOT 5,50
990 SETDOT 5,51 : SETDOT 5,52 : SETDOT 5,53 : S
ETDOT 5,54 : SETDOT 6,53
995 SETDOT 6,52 : SETDOT 6,53
1000 SETDOT 6,52 : SETDOT 7,49 : SETDOT 7,51 : S
ETDOT 7,51 : SETDOT 7,50
1005 SETDOT 8,48
1010 REM
1020 REM NEDRE NÄBBEN-NEDRE LÄGE TAS BORT
1030 CLRDOT 11,52 : CLRDOT 11,53 : CLRDOT 11,54
: CLRDOT 11,55
1040 REM
1050 REM NEDRE NÄBBEN-ÖVRE LÄGE SÄTTS
1060 SETDOT 9,52 : SETDOT 9,53 : SETDOT 10,54 :
SETDOT 9,54 : SETDOT 9,55
1070 FOR I=0 TO K : NEXT I
1080 OUT 6,0
1090 FOR I=0 TO K : NEXT I
1100 REM
1110 REM ÖVERNÄBBEN-NEDRE LÄGE TAS BORT
```

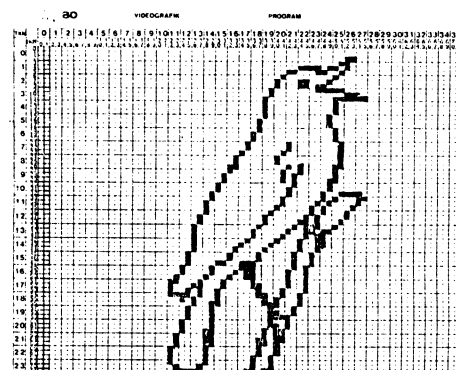


Bild 1c: Så kommer fågeln att se ut.

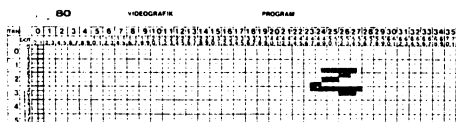


Bild 1d: Näbben får ett sångläge.

Insändare

Hej

Jag har beställt den utmärkta disassembleringen av ABC 80:s programvara. Där har dock smugit sig in ett fel bland 'labelarna'.

Felet finns på sidan 240 och det är log och log 10 som har förväxlat. Sidan 240 skulle ha sett ut såhär:

LOG	45:185	LD	BC,60:196
	45:188	JR	45:198
LOG 10	45:190	LD	BC,60:183
	45:193	JR	45:198

Genom att stryka över det gamla LOG 10 och fylla i 10 på det nya.

Felet beror troligen på att när man skulle fylla i 'labelarna' så förväxlade man dom, vilket är ganska lätt beroende på att dom ligger omvända i funktionstabellen.

Så här ser funktionstabellen ut:

.	
.	
.	
134	'INT KOD
INT	
136	'LOG 10 KOD
LOG 10	
135	'LOG KOD
LOG	
137	'PI KOD

Hoppas att det kan hjälpa någon som vill förstå logaritmbereäkningarna.

Hälsningar

2351 Mats Thulin

ÖNSKAS KÖPA

Flexskiveminne till ABC80
Datadisk, Luxor eller likvärdig.

Kurt Perman
Tallstigen 37
915 00 Robertsfors
<medl 1061>
0934-10989 efter kl 16

SÄLJES

ABC80. grundutförande, <11273>, med Facit skrivare säljes för 6.000:-
Tel 0418-15987

Facitskrivaren skriver på tryckkänsligt papper.

Hej !!!

Jag undrar om andra medlemmar som har ABC80 har samma problem som jag har? Problemet är att färgen på kåpan till tangentbordet lossnar. Det är inte bara min ABC80 som färgen har trillat av, jag har sett flera andra.
Det är säkert andra ABC-medlemmar som har samma problem.
Jag tycker att vi kan skriva till Luxor och klaga.

Lars-Ove Jonsson <2499>

SVAR

Efter årsmötet i februari 1981 vid paneldebatten där ledande personer vid Luxor Motala var närvarande kom denna fråga och Bengt Lönnqvist Luxor svarade då att felet låg i plasten kåpan är gjuten av. Detta har rättats till i senare tangentbordskåpor.
På frågan vad Luxor kunde göra åt problemet med gamla kåpor svarade Bengt Lönnqvist att dessa kunde sändas till Luxor Motala för utbyte mot ny kåpa som skulle tåla handslitaget bättre

Kanske bäst att ringa Luxor och fråga hur och vart att sända kåpan.

Reds Funderingar

Går man till en radio eller datorbutik för att KÖPA en -i denna fundering - dator och ska betala NY-pris vilket affären antagligen hoppas på - DÅ ska väl datorn vara OANVÄND !

Många i den uppväxande generationen har tydligen inte fått lära sig att inte pillra och peta på det mesta utan att betala för sig. "Prylarna står ju bara där, då gör det väl inget om vi använder dem", är en vanlig kommentar.

Ässå att vara med i någonting, exvis ABC-klubben, nästan så man hör 'kan vi inte få allt gratis utan att betala någonting'. Till synes förståndiga -vuxna- människor stod på årsmötet och pläderade för väldigt låga medlemsavgifter i klubben.
Vi är ju en ideell förening, men idealiteten får ju inte sträcka sig hur långt som helst.

Avgiften för juniormedlem är ju två - fyra biobiljetter - beror ju på var man bor och hur bra film - och sitter de ändå framför datorn har de ju snart sparat in till klubbavgiften.

Å kommer det insändare å de ska KNAPPAS in ska man inte behöva sitta å ilska till under inknappningen

claes s.

Falun 820912

Insändare till ABC-bladet

Fantasins nödvändighet - "Hachers"

Hur många missade denna underbara dokumentär om oss unga datoranvändare den åttonde september. Programmet var gjort av Anders Bergman huvudsakligen i Stockholm - Datorhuset, Solna m.fl. platser.
Jag är själv 16 år och går N1 i Falun, har hållit på med datorer sen 1979 nångång. Jag tycker programmet var jättebra gjort, det träffade mig verkligen.

När man kommer på någonting mitt i vardagen, och sedan söker efter första bästa lediga stund att testa idén på datorn. Fungerar det på en gång så blir man nästan förvånad - man glömmar alltid något utfall. Sen kanske man sitter med en likasinnad vid en dator nångång och den kommer med sina förslag, man testat och det hela kanske går bättre.

Får man så kontakt med mer erfarna programmerare - får se program som är mer avancerade, ex vis genom ABC-klubben, så vidgas ens vyer enormt. Man börjar upptäcka vad som finns i de innersta delarna av datorn - maskinkod. Därur kommer så småningom de bästa bitarna i programmeringen - assembler m.fl högnivåspråk. Nu är det tyvärr så att de flesta unga programmerare har mycket liten eller obefintlig kontakt med mer avancerade programmerare.

En del av skulden måste nog läggas på ABC-klubben. Vettiga åsikter om kraftigt reducerade priser för ungdomar framkom på årsmötet men tyvärr fick de inget gehör. Det är min förhoppning att åsikterna i denna fråga ändras till nästa årsmöte. Då vi förhoppningsvis beslutar om en ungdomsavgift på ca 25 kronor.

Tyvärr så såg vi vär knappast någon, möjligtvis en ABC80 i programmet. Det tycker i varje fall jag är ytterst dåligt. Programmet var ju helt svenskgjort och då bör man väl försöka få den stora SVENSKA datorn i stället för ca 98% Apple. Tyvärr måste nog Team 100/ABC80:s återförsäljare belastas för detta. Min erfarenhet av dessa är att de alltför ofta är avogt inställda till ungdomar som vill använda datorerna. Undantag finns naturligtvis, men på det stora hela tror jag tyvärr de är förhållandevis sällsynta. Man möts ofta med orden - "Det är för dyrbara grejor för er att hålla på med. Vi har inte tid." Nästan så man hör schas, schas efter sig. Så ser nan sen datorerna stå där dag efter dag och samla damm. Men fantasin är nödvändig både för barn och vuxna. Vi får så hoppas att få med --- fler unga användare i ABC-klubben. Ex.vis då genom kraftigt reducerade ungdomsavgifter och mera vänligt sinnade ABC80 återförsäljare.

Med en stilla förhoppning om en ljusare framtid

Martin Annerbo <2247> Falun

**Pausbird
kommer på kassett**

ABC V24

Körinvisning för ABCV24

ABCV24 som låg på ABC-kassett nr 3 är en specialutgåva av programmet T80PRT från Scandia Metric som ABC-klubben fick i julgäva 1981. På många begäran publiceras med beaktande tillstånd av Scandia Metric här en bearbetad körinvisning till programmet.

1. Läs in programmet från flexskiva eller kassett precis som vanligt.

2. Programmet presenterar sig med en julkänning från Scandia Metric. Därefter kommer på skärmen:

Detta program som gör V24-kontakten användbar för seriell ASCII, samt simulerar 80 kolumners bildskärm

Standardvärden (inom parantes) fås genom att trycka på RETURN-tangenten

Du kan välja mellan följande modul kombinationer

0.. Utskrift (ej ETX/ACK) (860 bytes)
1.. Utskrift (ETX/ACK) (1212 bytes)
2.. Utskrift och Inläsning (1432 bytes)
3.. Utsk + Inl. + Terminal (4712 bytes)
ETT ÖGONBLICK!
Välj 0 - 3 (3)?3
Standardprintertyp (KB1)?KB1
Max radlängd vid utsk (80)?80
Controlltecken (WEFVEPBÄ)?WEFVEPBÄ
1 eller 2 stoppbitar (1)?1
Enhetens namn (V24:)?V24:

Först frågas efter vilken modul kombination som skall användas (0, 1, 2 el 3).

(Genom att bara trycka RETURN erhålles för modul kombination 3).

3. Programmet frågar efter typ av yttre enhet. Tre olika parametrar skall anges: (H) (N) (B).

Vilket tecken som skall anges för respektive parameter beror på vilka konventioner som tillämpas av den yttre enheten (eller vilken typ av modem som används).

(H) Handskakning/paritet

Yttre enhet	Paritet/paritetsbit			
	MARK	SPACE	UDDA	JÄMN
SKRIVARE/- IN-ENHET (ej EXT/ACK)	A	P	B	C
SKRIVARE/- IN-ENHET Handskakning (med EXT/ACK)	E	D	F	G
ABC-80 som TERMINAL	I	H	J	K

Enbart en av dessa enheter skall anges.

(N) Antal null-karaktärer efter radframmatning (LF) / Half - Full Duplex
(null-karaktärer har ASCII-koden 0).

Yttre enhet	Parameter
SKRIVARE/- IN-ENHET	ABC-80 som TERMINAL
Inga null	Full duplex
2 null	Halv duplex
4 null	Halv duplex
6 null	Halv duplex
50 null	Halv duplex
	A
	B
	C
	D
	Z

Enbart en av dessa tecken skall anges.

(B) Överföringshastighet

Baud-tal	Parametrer
110	0
300	1
600	2
1200	3
2400	4
1200/75	5
75/1200	6

Enbart en av dessa siffror skall anges.

EXEMPEL:

När du ansluter en printer med 1200 Baud anger du t.ex CA3 och vid modem-kommunikation med ABC-klubbens Monitor KB1 (mot DEC-10 KA1).

4. Programmet frågar efter maximal radlängd. Svara med ett tal, som anger den radlängd som du vill ha på printern. Om du satt radlängden till 40 och skriver ut en rad som är 60 tecken lång, så skickar printer-rutinen ett extra <CRLF>, dvs Carriage Return + Line Feed när 40 tecken skrivits ut. Du får en utskrift som är skriven med "Wrap around" precis som på ABC80 bildskärm. (Genom att trycka enbart på RETURN erhålles radlängden 80). Genom att ange radlängd 255 så kopplas denna mekanism helt bort och du kan skriva ut hur långa teckensträngar du vill utan att få andra <CRLF> än du själv vill.

5. Programmet frågar efter önskade kontrolltecken. Svara med sju stora bokstäver som svarar mot de kontrolltecken man önskar använda. (Genom att enbart trycka på RETURN erhålles standardvärdena "WEFVEPBÄ"). För kontrolltecknens funktion, se nedan.

6. Programmet frågar efter antalet stoppbitar. Genom att bara trycka RETURN erhålles 1 stoppbit, vilket skall användas utom vid 110 baud.

7. Programmet frågar efter enhetens namn. Svara med en till tre stora bokstäver som skall vara enhetens namn, t ex PR:. (Genom att bara trycka RETURN erhålles namnet V24:). De flesta standardprogram använder enhetsnamnet PR:.

Därefter skrivs "ABC-80" ut och systemet är klart för användning.

OBSERVERA!

När du väl har valt modul kombination 0, 1, 2 eller 3 kan du utnyttja kombinationer med lägre nummer men inte med högre utan att läsa in ABCV24 på nytt. Du kan inte heller ändra max radlängd, enhetsnamn eller antalet stoppbitar.

Däremot kan du lätt byta kommunikationsparametrar. De kan nämligen också definieras vid OPEN, PREPARE eller LIST (av respektive fil). Detta kan vara önskvärt om man vill arbeta med fler än en fil, eller om man vill använda printer efter att ha kommunicerat som terminal.

Kommunikationsparametrarna är desamma, men går man tillväga på detta sätt blir formatet något annorlunda. Efter filnamnet skriver man (H)(N).(B). D v s, om parametrarna är KB1 skriver man: Filkommando "V24:KB.1" ASFILE filnummer.

Anm. ABC-80's kassettrutiner tar ej hänsyn till V24-porten. Detta medför att varje öppning, läsning eller skrivning på kassetten leder till att Null-karaktärer skickas ut på V24-porten. Denna utsändning stoppas först när man exekverar satsen:

OUT 58, INP(58) OR 8

eller när man öppnar V24-filen.

Terminalmodulen (3)

För att initiera ABC-80 som terminal måste man först öppna V24-porten på samma sätt som vid modul kombination (1) och (2) och därefter göra en INPUT-sats till samma fil. Sekvensen blir:

OPEN "V24:KB.1" ASFILE 1 (alternativt enbart V24:) INPUT \$ 1, A\$

När denna sekvens exekveras (det är exekveringen av INPUT-satsen som gör att ABC-80 intieras som terminal), antingen i ett program eller som ett direkt kommando intieras ABC-80 som terminal. Därefter fungerar ABC-80 precis som en vanlig asynkron terminal. Vill man avbryta detta arbets-sätt, d v s "koppla" ABC-80 så att den ej längre arbetar som terminal, trycker man ned CTRL-P på tangentbordet. Även datorn kan bryta förbindelsen genom att skicka ett bryttecken (vanligtvis CTRL-B) och därefter godtycklig text som avslutas med en RETURN-instruktion (CR), varvid texten mellan bryttecknet och CR hamnar i A\$.

Som BREAK-funktion, för att t ex bryta en pågående programexekvering, använder man CTRL-(E-accent).

Fortsättning från sidan 9.

Här är Fig Forth

Terminalrutinen ger ABC-80 en logisk bildskärm om 24 rader med 80 tecken per rad. Delar av denna kan visas på bildskärmen. Detta styrs av fyra funktioner:

Kommando ABC-80's bildskärm

CTRL-F Alla 80 kol för 12 översta raderna
CTRL-V Alla 80 kol för 12 understa raderna
CTRL-W De första 40 kol för alla 24 raderna stega 5 steg åt vänster
CTRL-E De sista 40 kol för alla 24 raderna stega 5 steg åt höger.

Dessa kommandon kan behöva definieras om, t.ex. man måste ha CTRL-W "ledigt" för att kunna skicka CTRL-W till datorn. Man kan då låta ett annat kommando ta över CTRL-W's uppgift vid styrning av bildskärmen så att CTRL-W blir "ledigt".

De kontrolltecken man önskar ändra kan då ändras vid uppstart av programmet i samband med att önskade kontrolltecken efterfrågas (se punkt 5).

Den logiska bildskärmen på 24 rader om 80 tecken per rad som terminalrutinen ger ABC-80 kan, vilket framgår av ovanstående, inte visas samtidigt på bildskärmen. Den lagras därför i RAMminnet. Förutom att använda ovanstående kommandon för visning av den logiska bildskärmen kan man även läsa av den från RAM-minnet med ett program. Beteckningen för detta 80-kolumners minne är SKR:.

Möjligheten att läsa den logiska bildskärmen från RAM-minnet gör att följande program kan användas för att skriva bildminnets innehåll på en printer, ansluten till ABC-bussen.

```
10 OPEN "V24:" ASFILE 1
20 OPEN "SKR:" ASFILE 2
30 OPEN "PR:" ASFILE 3
40 INPUT $1,$$
50 REM MANUELL BRYTNING M H A
   CTRL-P ELLER
60 REM DATORN BRYTER MED STX
70 FOR R=1 TO 24
80 INPUTLINE $2, A$ (R) REM LÄS
   BILDMINNET
90 PRINT $3,$$ (R) REM SKRIV UT PÅ
   PRINTER
100 NEXT R
110 GOTO 40
120 END
```

När det gäller användning av ABC-80 i kommunikation med ABC-monitor eller DEC-10 hänvisas till artiklar om ABC-Monitor i ABC-bladet.

Anvisning för inkoppling av printer och modem har också publicerats tidigare.

ABC-klubben vill passa på tillfället att rikta ett varmt tack till Scandia Metric.

Överföring från kassett till disk

För att lägga upp screens på disketten i det format de skall ligga i användes programmet DOSCREEN som ligger på ABC-kassett nr 4. Programmet överför screens från textfil SCREEN.TXT resp SCREEN2.TXT till Forth-format samt skriver ett directory med filnamnet "Forthscreen". Läser man det med vanliga LIB så kan man se hur många sektorer dessa screens tar upp. Forthscreen är ingen vanlig fil, man kan därför inte kopiera den med COPYLIB. För att kopiera till annan diskett måste man använda något program som kopierar från spår till spår i exakt samma form som originalet.

Så här går det till att från de levererade filerna på kassett göra en systemskiva för FORTH:

1. Överför först de aktuella programmen på ABC-kassetten till diskett med hjälp av programmet CASDISK.

2. Formatera en ny diskett och sätt den i drive 0.

3. Sätt skivan med de från kassetten överförda programmen i drive 1 och gör RUN DOSCREEN.

4. Svara på frågan om det redan finns ett screen-directory med tillräcklig plats. Svara N för nej när du gör en helt ny systemskiva.

5. Svara på frågan om det högsta screennummer du vill reservera plats för. RETURN sätter standardvärde (default) till 75.

6. Ange filnamn SCREEN.TXT resp SCREEN2.TXT.

7. Kopiera därefter över FORTH.ABS (eller FORTH32.ABS) samt CMDINT.SYS till den nya systemskivan med hjälp av COPYLIB.

8. Finns det plats kan du lägga in även andra filer t.ex. LIB, COPYLIB och COPYDISK som kan vara bra att ha på systemskivan.

Om du redan har en systemskiva med tillräcklig plats bör du svara J för ja i punkt 4. Frågan i punkt 5 hoppas då över och övriga programfiler kommer att vara oförändrade varför du inte behöver göra kopieringen enligt punkterna 7 och 8. Vill du utöka utrymmet för screens kommer alla gamla screens som ej skrivs över av nya att vara kvar oförändrade men du måste kopiera in övriga programfiler enligt punkt 7 och 8.

För att köra FORTH sättes systemskivan i drive 0 och du skriver BYE. Därefter ger du kommandot FORTH16 eller FORTH32 om du har extra minne och vill använda den versionen.

FORTH är nu klar att använda!

Ett MEN bara !!

ANVÄND UPPER CASE !!!
= Lampan skall lysa!!

Dokumentation till programvaran finns i ABC-klubben rapport nr 2: "Fig-FORTH on ABC-80, Implemented by Robert Johnsen", jfr Ref 3.

Rapporten som är skriven på engelska omfattar ca 50 sidor och innehåller bl a instruktionsmanual för att köra FORTH på ABC-80, screen-beskrivning, referenslista samt utdrag ur Fig-FORTH Installation Manual med Glossary (beskrivning av Forth Words) samt en beskrivning av Fig-Forth Editor. Rapporten kostar 60 kr totalt inkl frakt och beställes skriftligt från:

ABC-klubben, Vidängsvägen 1, 161 33 Bromma

och levereras mot postförskott. Det går också bra att betala in 60 kr på ABC-klubbens postgirokonto nr 15 33 36-3. Skriv Rapport nr 2 eller Forth-manual på talongen.

För den som vill lära sig FORTH rekommenderas varmt Brodies bok, se Ref 1. Beställes genom bokhandeln.

En kassettversion av FORTH beräknas komma på ABC-kassett nr 7. Det är vår förhoppning att programvaran skall komma att utvecklas och anpassas även till andra system såsom ABC-800 samt 8-tums flexskiveutrustning. ABC-klubben ställer gärna källkoden till förfogande till dem som vill göra sådana utvecklingar.

Gunnar Tidner

RÄTTELSE I FORTH

Programmet DOSCREEN

Rad 160: satsen efter kolon (:) skall bort.

Rad 260: nummret 255 utbytes mot 256.

GLÄDJANDE NYHET
för alla matematiker !!!!

Stig Rosenlund skrev ett brev att han skänker ABC-klubben sitt matematikpaket vilket han hade annons på i bladet 1-1982

Dessa program har anlänt, och kommer på kassett som vi hoppas inom en snar framtid.

Och när ni utvecklat kring-program med intressanta applikationer inom matematiken är Stig väldigt intresserad av en återföring över resultaten.

claus s.

DISKETT- och PROGRAMMÄRKNING

BLACKEBERGS GYMNASIUM
Datainstitutionen

Bengt Persson

BDS 100.0 utg 1 1982-05-21

BESKRIVNING AV NUMMERSYSTEMET

A. Programnummer

Programnummersystemet utvecklades i samband med inköpet av Blackebergs Gymnasiums första dator - en Alpha LSI-II - av Lennart och Sten-Axel Julin för att kunna hålla ordning på de gemensamma programmen i programbiblioteket. Detta var år 1973 och systemet har med smärre förändringar hållit i sig sedan dess - alltså nästan 10 år. Nu användes systemet på ABC80/800 med stor framgång.

Varje program gives ett fyrställt programnummer, där den första siffran anger ämnesområde/huvudgrupp och den andra siffran är en närmare specificering inom huvudgruppen. Exempel: Ett matematikprogram börjar på siffran 2, och om det rör sig om ett integralprogram börjar det på siffrorna 26. På detta vis går det alltid snabbt och enkelt att hitta något program inom det ämne man söker, förutsatt att man har en förteckning över sina program i nummerföljd.

De två återstående siffrorna användes till att skilja programmen åt. Eftersom man vid programmering har ett antal språk att välja bland, användes den 3:e siffran i programnumret till att beteckna program-språk. Kvar blir den sista som fungerar såsom skiljosiffra.

B. Diskettnummer

Vid användande av disketter som lagringsmedia finner man att ett tämligen stort antal program rymmes å varje diskett. Därför räcker det med 3-ställiga diskett-nummer. Första siffran anger alltså huvudgruppen, medan andra siffran betecknar programspråk (jfr programnumrets siffra tre) och den sista siffran fungerar såsom skiljosiffra.

Varje siffras exakta betydelse finnes redovisad i BDS 100.1 och BDS 100.2 till vilka hänvisas för närmare studium.

Synpunkter på nummersystemet mottages gärna under nedanstående adress:

Datainstitutionen
Bengt Persson
Blackebergs Gymnasium
Wergelandsgatan 16-24
161 58 BROMMA

BLACKEBERGS GYMNASIUM
Datainstitutionen

Bengt Persson
Sten-Axel Julin

1982-07-01 utg 5

BDS 100.1

Nummersystem för
program & disketter

PROGRAM-NUMMER

Bokstav efter
programnummer
betecknar under-
program.
Ex. 4171A

HUVUDGRUPP	Undergrupp	Programtyp	Skiljosiffra
0 Systemprogram	0	0 Program BASIC	0 Övergripande
1 Administration	1	1 " "	1
2 Matematik	2	2 " "	2
3 Ekonomi	3	3 " "	3
4 Fysik	4	4 " "	4
5 Kemi	5	5 " LISP	5
6 Teknologi	6	6 " Pascal	6
7 Grafik & Demo	7	7 " Assembler	7
8 Spel	8	8 Pgm fr. Pgm.lev.	8
9	9	9 Datafiler	9
A Biologi	A	A Program Algol	A
B	B	B " Simula	B
C	C	C " Comal	C
D	D	D	D
E	E	E Program Forth	E
F	F	F " Fortran	F
G	G	G " Cobol	G
H	H	H	H
I	I	I f.framt.expans.	I
J	J	J	J
K	K	K CP/M	K
L	L	L	L
M	M	M	M
N	N	N	N
O	O	O	O
P	P	P	P
Q	Q	Q	Q
R	R	R	R
S	S	S	S
T Textfiler	T	T Textfiler	T Textfiler
U	U	U	U
V	V	V	V
W	W	W	W
X Back-up-skivor	X	X	X
Y Museum D	Y	Y	Y
Z Övrigt	Z	Z	Z

D - endast
diskettmärkning

DISKETT-NUMMER

(se även BDS 102)

claes schibler knackade.

Samt för ABC-klubben:

Stig Löfgren
Måsvägen 16
183 51 TÄBY 08-756 57 20 H
756 57 30 A

BDS

BDS 100.2 utg 6 1982-05-02

NUMMERSYSTEM Program

0 SYSTEMPROGRAM

00 Allmänna systemprogram
01 Utskriftsprogram
02
03 Editorer
04 Kommunikation med datorcentral
05
06 Hjälpprogram till Pascal
07 Hjälpprogram till Assembler
08
09
0A Hjälpprogram till Algol
0B Hjälpprogram till Simula
0C Hjälpprogram till Comal
0D
0E Hjälpprogram till Forth
0F Hjälpprogram till Fortran
0G Hjälpprogram till Cobol
0P PEEK/POKE-hjälpprogram
0Z Övrigt

1 ADMINISTRATION

10 Allmänt
11
12
13 Ordbehandling
14
15
16 Skoladministration
17
18 Data- & registerhantering
19 Övrigt

2 MATEMATIK

20 Allmänt
21 Aritmetik
22 Matrisberäkningar
23 Geometri
24 Derivata
25 Differentialekvationer
26 Integraler
27 Funktioner
28 Komplexa tal
29 Övrigt

3 EKONOMI & STATISTIK

30 Sannolikhetslära
31 Kombinatorik
32 Befolkning
33
34
35
36
37
38
39 Övrigt

4 FYSIK

40 Allmänt
41 Laborationer NT1
42 Laborationer NT2
43 Laborationer NT3
44 Mekanik
45 Ellära & magnetism
46 Elektronik
47 Vågrörelselära & optik
48 Atomfysik & relativitetsteori
49 Övrigt
4X Liber Multilab

5 KEMI

50 Allmänt
51
52 Stökiometri
53
54 Kinetik
55 Syror & baser
56
57
58 Periodiska systemet
59 Övrigt

6 TEKNOLOGI

60 Allmänt
61 Materiallära
62 Tillverkning
63 Konstruktionselement
64 Mekanik
65 Hållfasthetslära
66 Ritteknik
67
68 Tabellvärden
69 Övrigt

7 GRAFIK & DEMO

70 TV-tennis, breakout o dyl
71 Andra bildskärmsorienterade spel
72
73 Färggrafik
74
75 Grafikdemo
76 Övriga demoprogram
77
78 Musikprogram
79 Övrigt

8 SPEL

80 Dungeons & Dragons
81 Kortspel
82
83 Brädspel
84 Krigsspel
85 Tärrningsspel
86 Hasardspel
87 Tankespel
88 Rymdspel
89 Övrigt

S BDS-FILER

S0
S1 BDS, Program- & Diskettnumrering
S2
S3
S4
S5 BDS, Programstandard
S6
S7
S8
S9

T TEXTFILER

T0 Datainstitutionen, allmänt
T1 Datainstitutionen, behörighet m.m.
T2
T3 Datainstitutionen, materialförv.
T4 Datainstitutionen, biblioteket
T5
T6
T7
T8
T9
T2 Datainstitutionen, övrigt

BDS 102 utg 4 1982-07-01

DISKETTIDENTIFIKATION

Diskettidentifikationen består av
6 tecken -- 3 bokstäver & 3 siffror.

De tre bokstäverna är ägarens signatur.
Några exempel:

BLA -- Blackebergs gymnasium
BPn -- Bengt Persson
HAS -- Harald Sjöstedt
NJB -- Jonas Bagge
PFE -- Per Ekman
SAJ -- Sten-Axel Julin

De tre siffrorna är diskettnumret.
Dessa väljes inte slumpvis, utan enligt
ett visst system, som är beskrivet i
BDS108.

Exempel på diskettidentifikationer:
BLA001 -- Blackebergs systemskiva
BPn800 -- Bengt Perssons spelskiva
(för Dungeons & Dragons)
NJB007 -- Jonas Bagges assemblerskiva

På så sätt kommer BLA006 att vara en
liknande skiva som BPn006. Detta har
visat sig mycket praktiskt, särskilt
då en skiva av någon anledning brutit
ihop.

BDS 108 utg. 2.1 1982-07-04

Nummersystem för DISKETTER

För ännu utförligare beskrivning se
BDS 110.

1. SIFFRAN (0)

0.. ALLMÄNT

00. Systemskivor (01. Systemskivor ABC800)

001 (010) Systemskiva 1
002 (011) Systemskiva 2
003 (013) Editering & utskrift
005 (015) LISP
006 (016) Pascal
007 (017) Assembler
008 (018) BASIC
009 (019) Procedurbibliotek BASIC
00E (01E) Forth
00F (01F) Fortran (brukas på egen risk)

02. Kommunikation, yttre enheter m m

020 Kommunikation via modem
022 Drivrutin, plotter
023 Kommunikation ABC80 (-M85)
024 Kommunikation M85 (-ABC80)

03. Systemskivor 8"

030 Allmän systemskiva

04. Systemskivor Metric85

040 Systemskiva
045 LISP
046 Pascal
047 Assembler
048 BASIC
04F Fortran
04G Cobol

05. -- 0G. Allmänna program i andra språk

än BASIC

0.0 Allmänt
0.9 Funktions-/ Procedurbibliotek

1.SIFFRAN (1 - Z)

1.. ADMINISTRATION
2.. MATEMATIK
3.. EKONOMI
4.. FYSIK
5.. KEMI
6.. TEKNOLOGI
7.. GRAFIK & DEMO
8.. SPEL
B.. BIOLOGI
T.. TEXTFILER

X.. BACK-UP-SKIVOR	0. Systemprogram
Y.. MUSEUM	01. Administration
Z.. ÖVRIGT	02. Matematik
	03. Ekonomi
	04. Fysik
	05. Kemi
	06. Teknologi
	07. Grafik & demo
	08. Spel
	09. Biologi
	0Z. Övrigt

Anmärkingar:

För de ovan inte närmare specificerade
2. & 3. siffrorna gäller följande all-
männa regler:

2. SIFFRAN

0. Allmänt
1. Program i BASIC spec f ABC800
3. Program i BASIC på 8" oavsett dator
4. Program i BASIC spec f Metric85
5. Program i LISP
6. Program i Pascal
7. Program i Assembler
8. Program i BASIC, originalsckivor
9. Datackivor

3. SIFFRAN

01 För ABC800
03 8"-skiva
04 För Metric85

BDS

BDS 515 utg 2 1981-07-07

Programhuvuds utseende

=====

BDS 515.1 BASIC-program

=====

```

1 REM +-----+
2 REM ! Program xxxx FILNAMN .BAC !
3 REM ! Utgåva xx 19xx-xx-xx !
4 REM ! signatur(er) !
7 REM +-----+
8 REM ! Kommentarsfält !
9 REM +-----+
10 REM
11 REM +-----+
12 REM ! Programbeskrivning ... !
13 REM !

```

```

! ! ! ! ! ! !
V V V V V V V

```

19 REM +-----+

Alternativt utseende på rad 4:

4 REM ! (c) Programförfattare !

Om rad 4 inte räcker till för programförfattarna användes dessutom rad 5 & 6.

BDS 515.2 Assemblerprogram

=====

TITLE 'FILNAMN.ASM'

```

; +-----+
; ! Program xxxx FILNAMN .ASM !
; ! Utgåva xx 19xx-xx-xx !
; ! signatur(er) !
; +-----+

```

I övrigt, se BDS 515.1

BDS 515.3 PASCAL-program

=====

PROGRAM FILNAMN;

```

(* +-----+
! Program xxxx FILNAMN .PAS !
! Utgåva xx 19xx-xx-xx !
! signatur(er) !
+-----+ *)

```

I övrigt, se BDS 515.1

BDS 520 utg 1 1981-11-19

Huvud för programavsnitt

=====

Varje programavsnitt inleds med en ruta, i vilken avsnittets namn anges samt viktigare variabler deklareraras.

Uppbyggnad:

```

xxx00 REM +-----+
xxx01 REM ! Avsnittets namn !
xxx02 REM +-----+
xxx03 REM ! X% - användning !
xxx04 REM ! Y% - användning !

```

```

! !
V V

```

```

xxx09 REM +-----+
(xxx19 REM +-----+)

```

BDS 526 utg. 1 1982-06-24

Enklare utformning av program- och

=====

programavsnittshuvud

=====

BDS 526.1 PROGRAMHUVUD, enkel utformning

=====

```

1 REM Program xxxw:yy PROGRAM.EXT

xxxx = Programnummer
w = Ev. bokstav efter programnummer;
    utfylles eljest med blank
yy = Utgåvenummer

```

BDS 526.2 PROGRAMAVSNITTSHUVUD,

=====

enkel utformning

=====

xxx00 REM === Avsnittets namn ===

(xxx99 RETURN)

Alternativ utformning av REM-satsen:

```

... Avsnittets namn ...
--- Avsnittets namn ---

```

BDS 530 utg 1 1981-06-25

Förteckning över standardprocedurer

=====

Nr Radnummer Procedurnamn Argument

```

9080 8000-8099 inputline Ut: G$
      G$
      G$

```

```

9252 25200-25299 kill In: F$
      Anm: Kräver 9281 & 9282. D$

```

```

9280 28000-28099 blankmask In: A$
      Ut: A$
      Anm: Blankmaskar endast 1. tkn

```

```

9281 28100-28199 filblankmask In: F$
      Ut: F$
      Anm: Främst avsedd för ABC800

```

```

9282 28200-28299 snyggt filnamn In:F$
      D$
      Ut:F$

```

```

9283 28300-28399 kort filnamn In: A$
      Ut: A$

```

```

9284 28400-28499 blankpet In: A$
      Ut: A$

```

```

9510 51000-51999 filfråga
      Anm: Kräver 9260, 9282, 9620
      och 9621.

```

```

9590 59000-59099 nytt utrymme Ut: B$
      Anm: Kräver 9621.

```

```

9591 59100-59199 frigjorda
      Anm: Främst avsedd för SYS.

```

```

9610 61000-61999 delete
      Anm: Kräver 9620, 9621 & 9622.

```

```

9620 62000-62099 ibiblängd Ut: L$( )

```

```

9621 62100-62199 utrymme Ut: B$

```

```

9622 62200-62399 filjämförelse In:F$
      Anm: Kräver 9625.

```

```

9625 62500-62599 filjämför
      Anm: Kräver 9252.
      Främst avsedd för SYS/DEL.

```

BDS 540 utg 1 1981-07-03

Förteckning över STANDARDFUNKTIONER

=====

Nr Funktionens utseende och funktion

```

9650 FNA$(Z$)=ASC(Z$) /ABC80/
      FNA$(Z$)=ASCII(Z$) /M85 & ABC800/

```

Avsedd för kompatibilitet mellan ABC80 & M85/ABC800.

```

9660 FNB0$(T$,S$)=CALL(24675$,SWAP$(T$)
      +S$*32$)

```

BLOCKUT till skiva.
T\$ - spår
S\$ - sektor

```

9661 FNB1$(T$,S$)=CALL(24678$,SWAP$(T$)
      +S$*32$)

```

BLOCKIN från skiva.
T\$ - spår
S\$ - sektor

9670 FNC\$(Z\$)=INSTR(1\$,A\$,Z\$)

Söker i strängen A\$ efter det tecken som medföljer som argument.

9730 FNI\$(Z\$)=INSTR(1\$,A\$,Z\$)

Söker iträngen A\$ efter argumentsträngen.

9740 FNJ\$=(FNA\$(G\$) AND 95\$)=74\$

Ger värdet 'SANT' om 1. Tecknet i strängen G\$ är 'J' eller 'j'.

9780 FNN\$=(FNA\$(/G\$) AND 95\$)=78\$

Ger värdet 'SANT' on 1. Tecknet i strängen G\$ är 'N' eller 'n'.

9810 FNQ\$(Z\$)=INSTR(1\$,A\$,CHRS(Z\$))

Söker i strängen A\$ efter det tecken vars ASCII-värde angivits som argument.

TILLÄGG till artikeln
Annorlunda Stränghantering i nr 1. 1982
Följande program tillhör Fjärde Provet
sidan 31

```

10 A$='*****'
90 P=32768 : REM ...save KOD....
83 03 03 SW
95 ; 'TITTA/SKRIVA (T/S)'; : GET
W$ : ; W$
97 IF W$='S' THEN 300
100 A=PEEK(P+I)
110 ; P+I;I,A,CHRS(A)
120 GET W$
130 I=I+1
140 GOTO 100
200 REM ...SAVE KOD
300 FOR J=0 TO 4
310 READ A
320 POKE P+12+J,A
330 NEXT J
340 ; 'DATA INSKRIVNA'
350 GOTO 95
400 DATA 33,34,35,36,37
500 RETURN

```

SÄTT AUTOPILOT PÅ DIN DIN ABC80!

Användarrapport

Nyfors Data Science erbjuder ett programhjälpmedel som man kallar Superbasic. Bakom firmanamnet finns Magnus Lundberg och Magnus Jacobsson. Jag har haft tillfälle att prova ett par varianter.

En huvudidé bakom Superbasic är att ABC80:s BASIC får kommandolistan utökad med ytterligare ett tjugor ord som kan beställas direkt från tangentbordet.

Många av dessa kommandon är kända från andra i marknaden förekommande produkter - AUTO (automatisk radnumrering), DEL (ta bort önskat antal programrader), VAR (visa använda variabler) - men funktionerna är ofta utökade.

Med VAR X visas alla rader där variabeln X förekommer - ovärderligt vid programbearbetning.

Ibland vill man ha tag på något som inte är en variabel, t ex REM, END eller en textsträng. Då finns kommandot FIND som gör jobbet på samma sätt som VAR.

LIB och COPY kan utföras direkt utan att påverka arbetsminnet.

Med den inbyggda och alltid inkopplade programeditorn kan markören flyttas till valfri plats på skärmen. Programrader kan editeras, för långa rader delas upp, felaktigt skrivna rader "räddas".

Med VISA <namn> hämtas BASIC-filer och textfiler från flexskiva och visas på skärmen utan att påverka arbetsminnet. Man kan på det sättet hämta rutiner från andra program och kopiera in dem i det program man har just då i arbetsminnet.

Superbasic innehåller en HOLD-funktion som är mycket användbar: shift+CTRL+bokstaven O "frysar" programexekveringen i de flesta program. Ett andra tryck på samma tangent släpper loss igen. Om man i stället ger kommandot CTRL-P sker utskrift på printer av skärminnehållet.

Ibland händer oväntade saker och program låser sig. Enda boten har varit RESET och förlust av programmet. Nu kan man efter "frysning" välja CTRL-B, vilket för det mesta frigör låsningen - och RAM-minnet förblir opåverkat. Den funktionen har man ofta anledning att väljsigna!

Om man av misstag gjort NEW och blivit av med sitt program, kan man med OLD återställa pekarna och jobba vidare.

Det finns fler små nyttiga och roliga funktioner. Men här finns tre programidéer som är alldeles speciella och mig veterligt inte lanserats förut: STEP, EDIT och EXEC.

Med STEP kan ett BASIC-program exekveras stegvis. Varje programrad visas på skärmen innan den utförs. Man kan rätta till upptäckta fel och fortsätta exekveringen med STEP<radnr>. Ofta är detta praktiskt när man prövar en rutin eller letar efter underliga fel.

EDIT är ett ordbehandlingsprogram, eller åtminstone början till ett. Vi har ju tidigare ABC-ORD och TV, och dessa båda har finesser som inte EDIT har. Men EDIT har två intressanta fördelar. För det första finns programmet alltid tillgängligt under hela den tid Superbasic är inkopplad. För det andra kan man behandla i det närmaste obegränsat långa textfiler.

Med kommandot EDIT\$<programnamn> skapas en textfil. På skärmen skriver och editerar man sedan sin text. Räknare för bl.a. rad och kolumn visas på skärmen och hjälper till att hålla koll på storleken.

När man är klar ges kommandot CTRL-Q samt EN (för end) och filen skrivs in på flexskivan.

Om arbetsminnet blir fullt innan dess, kan texten efter hand överföras till skivan, men man får då kvar i minnet en skarvbit, så att man kan hålla reda på var man är.

Om man vill editera en redan befintlig fil fungerar det på samma sätt. Programmet läser in så mycket som får rum i minnet. När man arbetar sig genom texten flyttas efter hand överskottet till skivan igen. Vid avslutningen kan man välja om man vill spara den gamla filen (under extension .BKP) eller bara den nya. Det är alltså bara skivans kapacitet som anger gränsen för hur stora textfiler man kan jobba med.

EDIT är inte färdigutvecklat och kommer nog att förbättras åtskilligt i kommande upplagor. Programmet fungerar bäst på 80-teckensskärm. Den version jag har provat visade upp en del barnsjukdomar och tycks inte vara riktigt pålitligt i min maskin. Klart är att programiden har lovande utvecklingsmöjligheter.

Nu har det bästa sparats till sist: EXEC, som kan fungera som autopilot på ABC80! I broschyren står att du "kan arbeta i BATCH med JOBSTRÖMMAR där du har GLOBALA PARAMETRAR". Min första tanke var att detta är något för de stora grabbarna och inte intressant för mig. Efter att ha provat har jag ändrat åsikt. Med EXEC kan man i början av körningen och på en gång ge datorn alla de kommandon, instruktioner och frågor som behövs för att hålla den i gång ända tills hela jobbet är avslutat.

Ett exempel: Anta att du har fem textfiler som du vill ha utskrivna med utskriftsprogrammet SKRIV, i vilket du vill välja olika värden på radlängd, marginal osv. I vanliga fall får du sitta vid tangentbordet och vänta tills maskinen är färdig med den första texten, innan du kan ge nödiga instruktioner för den att köra nästa text.

Med EXEC skriver du genast ned alla de instruktioner som behövs för jobbet. Sedan kan du gå hem och låta apparaten jobba. Det är också möjligt, för tex återkommande jobb av samma slag att spara jobströmmen i en specialfil. Denna kan t o m kalla in sig själv med olika värden och upprepa sig så många gånger som behövs.

Jag blev helt förbluffad över vad min lilla ABC80 plötsligt kunde åstadkomma. Även om man har en del fallgropar (bl a måste varje kommando vara helt rätt och alla måste komma i rätt ordning) är jag ganska säker på att det kommer att vara en användbar funktion.

Hur får man nu in denna Superbasic i sin ABC80? Magnus Lundberg erbjuder varje kommando i en särskild modul: köp och betala bara de kommandon du vill ha! Den som nöjer sig med bildskärmseditorn, AUTO, VAR, FIND, DEL, VISA, LIB och några till slipper undan med ca 500 kr och kan få alltsammans på en flexskiva. Superbasen läses in i RAM-minnet och läggs under taket där den vanligen inte stör. Men den upptar i alla fall något tusental bytes av arbetsminnet.

Skall man ha alla kommandon blir det dyrare. EDIT och EXEC kostar tillsammans 700 kr. De tar också väl mycket plats, även om man väljer att lägga dem i några av dosbuffertarna.

Magnus rekommenderar en dyrare men bättre lösning: Supern läggs i ett par PROM-kapslar på ett expansionskort. Kortet pluggas in på en ledig plats i diskettläsaren. Då finns alltsammans där hela tiden och lämnar det ordinarie arbetsminnet orört!

Jag har provat båda möjligheterna. Den lilla lösningen är utmärkt att börja med när man inte har så gott om pengar. Men min erfarenhet var att jag snabbt insåg mitt behov av fler och fler rutiner. Lösningen med PROM-kort är utmärkt, men tyvärr dyrt - kortet kostar också.

Just när detta skrives har jag fått veta att man experimenterat fram ytterligare en metod att placera PROMarna så att man inte behöver något expansionskort. Den har jag inte kunnat pröva.

En sista slutknorr: Om man byter en DOS-kapsel i diskettläsaren mot en specialare (kostnad 500 kr) får man tillgång till ytterligare några hjälpmedel som kan vara bra att ha: Dagens datum läses in på en ledig uddaplats i minnet och skrivs sedan automatiskt in på alla nyskapade eller ändrade filer. Dessutom läggs både datum och filstorlek in i biblioteket och visas direkt på skärmen med kommandot BIB (som ersätter LIB).

Det nya DOSet kan kombineras med automatisk programstart och gör även ett par andra saker som man inte direkt märker, men som snabbar upp hanteringen och gör den tillförlitligare.

Sven Wickberg

UTFÖRANDE AV KOMMANDON I BASIC ELLER I MASKINSPRÅKSPROGRAM

Av Mika Helenius
(översatt från finska av Risto Koivula)
(källa: Finska ABC-bladet 2/81)

Nedan följer en enkel Basic-rutin, som genom POKE med början på adress -128 sänder strängen Cx + carriage return.

```
10000 C$=C$+CHR$(13%):FOR Z%=1%
TO LEN(C$):POKE -129%+Z%,
ASC(RIGHT$(C$,Z%)):NEXT Z
%:RETURN
```

KOMMANDO	ADRESS HEX	ADRESS DECIM.	FINNS PARAMETER	PARAMETER INNEHÅLL
BYE	017A	378	-	
LIST	0AEF	2799	+	filnamn
LOAD	0BA8	2984	+	filnamn
MERGE	0BA9	2985	+	filnamn
NEW	0A70	2672	-	
RUN	0D4C	3404	+	filnamn
RUN	0D55	3413	-	vanligt
RUN	0D58	3416	-	utan CLEAR
SAVE	0D6C	3526	+	filnamn
UNSAVE	014C	332	+	filnamn

Det händer ofta när man gör ett program, att man önskar ABC-basinen gav möjlighet att utföra kommandon i programmet. Kommandon som RUN, SAVE, LIST, MERGE och NEW skulle i många program vara användbara och ibland direkt behövas. Så gott som alla dessa kommandon kan dock användas med en lämplig CALL-sats.

Alla ovanstående kommandon kan anropas via CALL, såväl i Basic som i assembler. Kommandon utan parameter är lättast att använda. Där räcker det med ett maskinspråksanrop, t.ex.:

```
999 Z%=CALL(3413):REM RUN
```

startar programmet från början, nollställer alla variabler samt stänger alla filer.

Kommando som kräver parameter är litet knepigare. Då fordrar systemet att HL-registerparet innehåller adressen till början på parameterlistan. Parametern måste avslutas med ASCII-tecknet 13 (carriage return). Problemet är att man inte kan sätta HL-registerparets värde vid CALL-anrop.

Nedan en maskinspråksrutin som laddar HL-registerparet med värdet -128 (parameterns startadress) och därefter hoppar till adressen som finns i DE-registerparet. Eftersom DE-parets innehåll kan bestämmas via ett CALL-anrop, erbjuder dessa rutiner ett sätt att enkelt utföra kommandon.

```
1 POKE -9,213,33,128,255,201
```

Om man antar att rad 1 är körd utförs kommandot

```
LIST 100,300
```

på följande sätt:

```
10 C$="100,300":GOSUB 10000
15 Z%=CALL(-9,2799):REM LIST
```

Genom att i stället för talet 2799 skriva adressen för respektive kommando, kan man utföra vilka kommandon som helst, t.ex. filnamn i samband med RUN-kommandot, eller helt tom.

Om parametern är felaktig, får man under körningen fel, som inte tas om hand av ON ERROR GOTO satser utan programmet stoppas.

Några användbara tips plockade ur den finska ABC-klubbens tidning nr. 1 & 2 -81.

Efter det att ett program stoppats genom ett felmeddelande eller genom CTRL-C, kan man fortsätta programmet utan att förändra variablerna genom kommandot:

```
POKE 65060,0:GOTO radnummer
```

Kommandot måste ovillkorligen ges på samma rad. Programmet kan inte fortsätta från en subrutin eller från en FOR - NEXT loop. Adressen 65060 indikerar körtillstånd (0) eller kommandotillstånd (1).

```
DEF FNM(X,Y)=-X*(X>Y)-Y*(Y<=X)
DEF FNN(X,Y)=-X*(X<Y)-Y*(Y>=X)
```

Funktionen FNM returnerar största värdet av X och Y, FNN returnerar det minsta. Dessa funktioner är speciellt användbara vid anpassning av HP-2000 program till ABC80 och ersätter då funktionerna MIN och MAX.

Vad gör följande program?? Gissa..

```
1000 POKE -128%,243%,201%
1010 Z%=CALL (-128%)
1020 ; CHR$(PEEK(-528%));
1030 IF INP(56%)<>131% 1020
1040 ; "TRODDE DU.."
1050 GOTO 1030
```

Pröva..

SÄLJES

Radskrivare Scandia-Metrics ITOH med parallellsnitt o engelsk bruksanvisning.

Sven Wickberg 0750-50456 (kvällstid)
0750-50494 (dagtid)

SUPERBASIC

Utökar ABC80's basictolk med 8KB som därmed blir det oslagbara programmeringshjälpmedlet. Din ABC80 blir som en stor dator som kan arbeta i batch med jobbströmmar och globala parametrar. Allt som normalt skulle matas in från tangentbordet tas i stället från jobbströmmen, som kan vara en i förväg preparerad fil eller ett spontant kommando. En inbyggd skärmorienterad texteditor för alla slags texter såsom brev, assemblerprogram etc finns alltid till hands och aktiveras med ett enkelt kommando. Vid mycket långa rader scrollas skärmen i sidled. Obegränsad textlängd. Singel step-funktion som före exekvering av en basicsats visar hela raden nederst på skärmen ger möjlighet att testa basic-

program radvis. Det går att starta ett program från valfritt radnummer med bibehållna variabler. En 'continue'-liknande funktion.

Dagens datum underhålls i systemet. Med ett litet tillägg i DOS'et går det även att automatiskt få filer markerade med dagens datum då de uppdateras. Med BIB-kommandot visas filernas senaste ändringsdatum.

Inbyggda kommandon för kopiering, lib, de-/kryptering, komprimering, automatisk radnumrering, sökning av variabler och strängar och mycket annat smått och gott. Alla kommandon både ABC80- och SUPER-kommandon kan även utföras från program.

SUPERBASIC fungerar med eller utan TKN80 i alla basictolkar. SUPERBASIC kan konfigureras efter kundens önskemål, men levereras som standard i en lätt utbytbar EPROM i en liten låda som sätts i busskontakten på tangentbordet. Med varje leverans av SUPERBASIC följer en utförlig beskrivning och en demonstrationsdiskett med exempel.

Till Nyfors Data

☐ Jag beställer Superbasic

☐ Jag vill ha mer information

Namn:

Adress:

..... Tel:

Kortlek

Datorn används ofta för simuleringsuppgifter. Här skall vi se hur man simulerar en kortlek.

Det finns flera olika sätt, de flesta bra, men det snabbaste och enklaste tycks vara att använda siffervärden för korten.

Låt 100 betyda en färg, 200 nästa osv, och låt kortens valörer gå från 1 till 13. Då kan 101 vara såg hjärter ess, 213 spader kung etc.

Låt sedan varje kort få variabelvärdet A(x),

Detta görs i programmet KORTLEK på raderna 100-140. Rad 125 är en onödig kontrollrad. Den visar bara att det har blivit rätt. Skriv in t o m rad 140 och kör, så ska du få en snygg listning i nummerordning av de i det föregående definierade korten, dvs 101-113, 201-213, 301-313 och 401-413.

Nu gäller det att blanda och ge. Det görs i 200-260. Vi har från början 52 kort (N) och drar därför ett slumpstal 1-52 (220). Om du ändrar rad 230 till ;A(X); får du nu en utskrift av samma tal som ovan, men i annan ordning. (För att det inte skall bli samma ordning varje gång man börjar spelet, måste du ha RANDOMIZE någonstans i början av programmet, t ex rad 70.)

När man dragit ett visst kort vill man ha bort det ur leken. Det sker med samma metod som lottdragningen HATTEN i nr 4/5-1982. Man tar det "understa" kortet och lägger på platsen X. (Vill man ha det dragna kortet kvar i leken får man göra ett ordentligt byte - C=A(X);A(X)=A(N);A(N)=C. Då har de båda korten bytt plats.)

Därpå minskar man N med 1. Nästa gång sker dragningen av ett tal 1-51, dvs bland de nu återstående korten.

Därmed skulle i stort sett uppgiften vara löst. Men man kanske gärna vill tolka siffrorna till riktiga kortnamn? I vårt program gör vi det i subrutinen 300 och framåt.

F i 330 blir hundratalssiffran, som anger färgen. V i 340 blir de två sista siffrorna som innehöll valören. Därmed översätts F till klartext, liksom V för ess, knekt, dam och kung. Med några mindre ändringar kan detta program ingå som subrutin i sådana program där man använder kort.

Några pedagogiska randanmärkningar. Egentligen skall man inte skriva villkorssatser så som i 350 och framåt. Varje IF-sats borde avslutas med :GOTO 420 och därmed hoppa över alla andra villkorstester. Då skulle alla korten med valör 2-10 självklart hamna i sista raden, och då skulle man inte behöva villkoren på denna rad. Det samma gäller färgerna, där man efter "napp" omedelbart skulle slussas till rad 370.

I detta fall har jag syndat på nåden, eftersom programmet ändå går så fort och det blir enklare att skriva ut.

Vill man minska minnesutrymmet och öka hastigheten (kan märkas om man t ex skall dra 5 kort åt 3 deltagare), rekommenderas att alla siffervariabler görs till heltal.

Sven Wickberg

Personnummer

```

10 REM list PNR
1000 REM ..INMATNING
1010 ; 'Skriv personnummer:': ; I
      NPUT W$
1020 IF LEN(W$)<10 OR LEN(W$)>11
      THEN 1010
1030 W1$=LEFT$(W$,6)+RIGHT$(W$,8)
      : REM ta bort bindestrecket
1040 ONERRORGOTO 1010 : W=VAL(W1$)
1500 REM ...BERÄKNING
1510 FOR I=1 TO 9
1520 A$=MID$(W1$,I,1)
1530 A=VAL(A$)
1540 IF I/2<>INT(I/2) THEN A=A*2
1550 IF A>10 THEN A=A-9
1560 B=B+A
1570 NEXT I
2000 REM ...koll om rätt
2010 B=B-INT(B/10)*10
2020 K=10-B : IF K=10 THEN K=0 :
      REM kontrollsiffran
2030 IF LEN(W$)=10 ; 'Kontrollsisf
      fran skall vara:':K : END
2040 K$=MID$(W1$,10,1) : K1=VAL(K
      $)
2050 IF K1=K ; 'Numret är rätt!'
      ELSE ; 'Numret är fel!'

```

Som bekant ha alla i Sverige skrivna personer ett identitetsnummer bestående av 6 siffror för födelseår,-män,-dat och därtill ytterligare 4 siffror. De två första av dessa har att göra med det län där man föddes eller första gången mantalsskrevs. Nästa siffra används dels för att skilja män och kvinnor (udda för män), dels för att särskilja sådana som har alla övriga siffror gemensamma, t ex pojkar som föds samma dag på samma plats. Då får den förste siffran 1, nästa siffran 3 osv. (Hade det varit flickor hade de fått resp 0, 2 osv.)

Den tionde och sista siffran är en s k kontrollsiffra, som används för att kontrollera att personnummret är riktigt.

Hur går det till? Man multiplicerar varannan siffra med 2 och varannan med 1. Alla siffrorna i dessa produkter läggs ihop. Sista siffran i den nya summan dras från 10. Då har man fått kontrollsiffran.

En figur gör det möjligen mera begripligt:

```

personnummer:  5 8 1 1 2 3 - 1 4 0
multipl. med:  2 1 2 1 2 1 2 1 2
-----
vilket ger   : 10 8 2 1 4 3 2 4 0

```

siffersumman: 10+8+2+1+4+3+2+4+0=34
sista siffran i siffersumman är 4

kontrollsiffran är 10-4=6

Vad skall nu detta vara bra för? Experterna har räknat ut att om man skriver en siffra fel eller kastar om två siffror, är chansen mycket liten att kontrollsiffran ändå stämmer. Man får alltså en kontroll på att personnumret är rätt.

Är det någon som gissat att detta är ett typiskt datorjobb? Rätt i så fall. För datorn är det en enkel match att räkna ut det här.

Programmet PNR visar hur det kan göras.

Programmet förväntar sig en inmatning på 9 eller 10 siffror plus bindestrecket. Efter sedvanlig rimlighetskontroll tas bindestrecket bort.

Rad 1040 kollar att återstoden är numerisk (bara innehåller siffertecken), och sedan tar man fram tecken för tecken, förvandlar till tal (1530), multiplicerar med 2 om 1 är udda (1540).

Sedan skall siffersumman tas av produkten. Eftersom produkten kan vara högst 18 (dvs 9x2) kan man göra det litet enklare för sig (1550).

I B adderas alla summorna upp, och i 2010 görs den vanliga manipulationen för att ta bort tiotalssiffran.

Denna rutin kan putsas något och ingå som subrutin i sådana program, där man har behov av kontroll av personnummer.

Om man arbetar med heltal istället spar man litet tid och litet plats.

Sven Wickberg

```

10 REM ...SAVE KORTLEK...82 04
      04 SW
80 REM P=1 : OPEN 'PR : ' ASFIL
      E 1
90 DIM A(52) : N=52
100 FOR I=100 TO 400 STEP 100
110 FOR J=1 TO 13
115 K=K+1
120 A(K)=I+J
125 ; A(K);
130 NEXT J
140 NEXT I
150 GET W$
200 REM Blanda och ge
210 FOR I=1 TO N
220 X=INT(52*RND)+1
230 GOSUB 310
240 A(X)=A(N) : N=N-1
250 NEXT I
260 END
300 REM Klartext
310 F=INT(A(X)/100)
320 V=A(X)-F*100
330 IF F=1 THEN F$='HJÄRTER'
340 IF F=2 THEN F$='SPADER'
350 IF F=3 THEN F$='RUTER'
360 IF F=4 THEN F$='KLÖVER'
370 IF V=1 THEN V$='ESS'
380 IF V=11 THEN V$='KNEKT'
390 IF V=12 THEN V$='DAM'
400 IF V=13 THEN V$='KUNG'
410 IF V>1 AND V<11 THEN V$=' '+
      NUM$(V)
420 ; $P,F$,V$,
430 RETURN

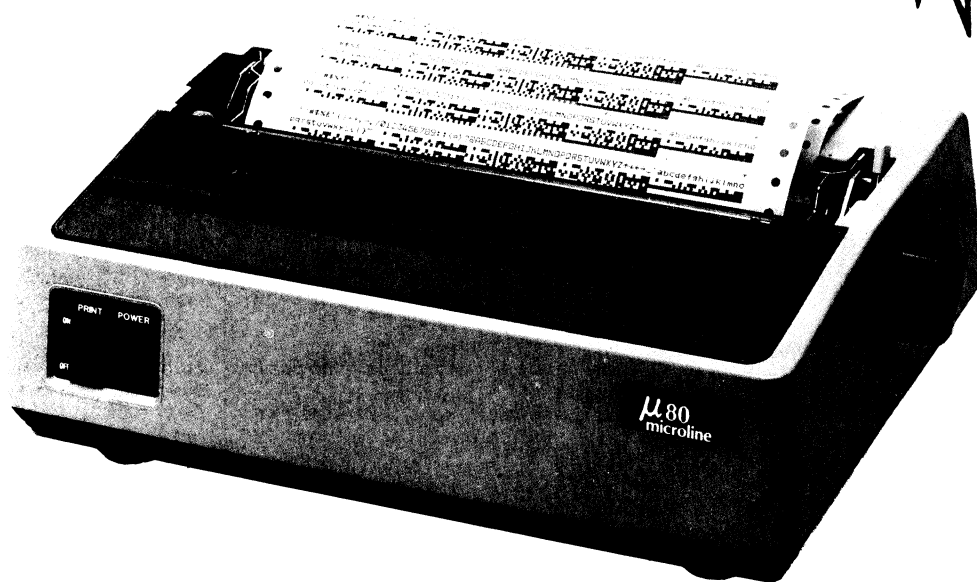
```

Medlemserbjudande!

Microline 80

Serial Dot Matrix Printer

80 cps Unidirectional 80 Column



- * HÖG KVALITET - LAGT PRIS
- * SKRIVHUVUDETS LIVSLÄNGD 200.000.000 TECKEN
- * VANLIGT FÄRGBAND
- * 40, 80 ELLER 132 TECKEN PER RAD
- * VANLIGT PAPPER + TVÅ KOPIOR
- * SVENSKA TECKEN
- * ABC 80 GRAFIK

10% RABATT PÅ ÖVRIGA
MICROLINE MODELLER.

	Parallell	Seriell
ML 80 printer	3.639.-	3.639.-
Int.face+kabel	1.209.-	1.324.-
	4.848.-	4.963.-
RABATT 20%	969.-	992.-
DITT PRIS Komplett	3.879.-	3.971.-

Alla priser inkl moms.

Uppge medlemsnummer vid order !

**MAGNIUS
ELEKTRONIK**

LUNDSGATAN 13
260 33 PAARP

TELEFON:
042 - 22 73 20

Stortext

Man har ibland behov av att skriva med större text på datorskärmen än den som är standard. För detta finns ett eller annat program på marknaden, t ex MINDEMO av "the mad programmer" Per Lindberg

Alla har det gemensamt att man skriver bokstäverna med hjälp av en matris, som lagras i datorn med koordinaterna för rad och kolumn.

I MINDEMO innebär detta en formidabel räckvidd av omfångsrika DATA-satser, som stjäls massor med minne och är mer än lovligt bökiga att knappa in (den som gjort det har också skrivit en kommentar om detta längst ned!).

I ABC-bladet nr2/1981 (sid 22) finns ett uppslag av Niklas Nottelman i programmet BOKSTAV, att i stället spara matrisen i form av sju binärtal (i decimal form). Då behöver man bara sju tvåsiffriga tal i DATA-satsen för varje bokstav.

Vitsen är att binäretter ger fylld ruta, medan binärnollor ger tom ruta. På det sättet bestäms rad för rad vilka rutor i matrisen som är fyllda och vilka som är tomma. Matrisen fylls mycket snabbt radvis.

I närskrivna program STORTEXT demonstreras en annan id, som kommer från Gunnar Hasselberg och bearbetats av en smula. Den bygger på samma matrisid som Per Lindberg, men i stället för att ange rad och kolumn som ett tvåsiffrigt tal, och ha komma mellan varje sådant tal, registrerar man talen som tecknet med motsvarande ASCII-värde.

65 skrivs alltså som A, 99 som c osv. Fördelen med detta är att man behåller intrycket på skärmen av att bokstäverna "skrivs" fram, vilket ger känsla av "action", samtidigt som man högst avsevärt kondenserar det behövliga minnetrymmet. Det blir också bekvämare att skriva in programmet, och listan blir mycket mindre omfångsrik.

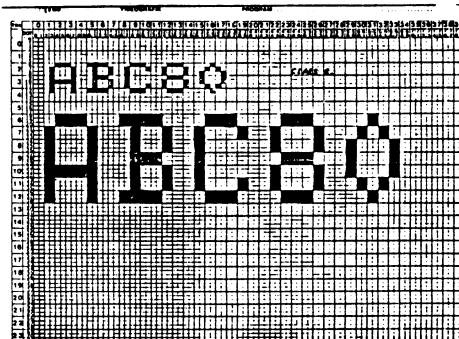
I vårt exempel, som bara visar själva id'n, ges bara tre bokstäver. För att få ASCII-värden inom ett lämpligt område har matrisen fått radnummer 60-120 och kolumnnummer 1-5.

Vill man vara litet vidlyftigare och i stället har en 9x5 matris (för att få med under-slingar) kan man ju ta med raderna 40 och 50.

Själva utskriftsslingan behöver också förbättras så att den blir generell och skriver texten i snygga rader, men den övningen överlätes åt den intresserade. Här visas bara två mycket enkla modeller att ta fram fyllda rutor, med CTRL-> (=ASCII 127) på rad 1200 och med SETDOT på rad 1300.

På rad 1100 har använts R% och K% (heltal) i stället för INT(1/10) osv. I princip bör man ju använda heltal överallt där det går i ett program som detta, eftersom datorn skall 'arbeta själv' så mycket mellan vändorna på tangentbordet, att tidsvinsten blir märkbar.

Sven Wickberg



```
100 REM ...STORTEXT
200 REM Id` av Gunnar Hasselberg
210 REM presentation av Sven Wickberg
215 REM Jobbar med ett rutnät där
220 REM övre vänstra rutan är 61
225 REM raderna är tiotalen R%
230 REM kolumnerna är entalen K%
240 REM Talen symboliseras av ASCII-
250 REM koderna hos motsvarande tecken
260 REM L anger förflyttning i s
idled
270 REM mellan bokstäver
280 REM Fattas rutin för radförflyttning
290 REM Layouten kan förbättras
300 N=5 : REM antal tecken att skriva
400 : CHR$(12)
500 FOR I=1 TO 23 : ; CHR$(151)
: NEXT I
600 FOR J=1 TO N
700 READ A$
900 FOR I=1 TO LEN(A$)
1000 A=ASC(MID$(A$,I,1))
1100 R%=A/10 : K%=A-10*R%
1200 : CUR(R%,K%+L)CHR$(127%)
1300 SETDOT R%,(K%+L)+2
1400 NEXT I
1500 L=L+7
1600 NEXT J
1700 : CUR(19,5)'KLART'
1800 DATA yoeÄQG>?'KU isäfgh
1900 DATA =>HRöfpzy?'KU Äisöäz
2000 DATA K'?'>GQÄeoäös
2100 DATA '?>GQÖÄ^isöäzoeUK
2200 DATA ?HQÄepäri_UJ
```

ENDAST FÖR FLOPPYISTER

Vill du få dubbelt utrymme på dina flexskivor? Det finns ett enkelt sätt: utnyttja även baksidorna!

De flexskivor man köper i dag är mestadels likadana på båda sidor. Anledningen till att man endast kan använda ena sidan är det lilla hålet intill det stora centrumhålet (se skiss). När skivan sätts in i skivspelaren måste drivmekanismen ta tag i det stora centrumhålet och läsning/skrivning ske genom det ovala hålet nedtill på skissen. Dessa hål ligger symmetriskt kring mittlinjen (streckad) och fungerar lika bra även om man vänder skivan kring mittlinjen.

Det lilla hålet motsvarar ett ännu mindre hål på själva skivan. Genom dessa hål skall en ljusstråle kunna lysa när skivspelaren är i funktion. Denna del av mekanismen sitter i de flesta enheter nedanför det stora hålet i skivan. Vänder man skivan, kan ljuset inte komma igenom.

Det botas dock lätt om man gör ett motsvarande hål (dvs två - ett på var sida om skivan) i skivfodralet som skissen visar. Detta är inte fullt så enkelt som det låter. Hålen på vardera sidan måste ligga på samma ställe och passa ljusstrålen i apparaten.

Man kan med en vass sax klippa upp en fyrkant på var sida kring det aktuella stället. Det finns risk att man därvid skadar den del av skivan där dataspårerna skall finnas. Om skivan är tom gör det inte så mycket om något spår skadas: det tas bort vid formateringen. Även om man skulle förlora fem spår på varje sida, vinner man 35 spår på baksidan.

Våra tekniskt sinnade läsare kan säkert komma på något smartare sätt att göra hålet. Hör gärna av er!

Medan saxen ändå är framme måste man också klippa ett jack i kanten - den som kommer att vara nedåt när man vänt skivan. Annars kan man ju bara läsa och inte skriva på den nya sidan.

Varsågod, operationen klar och du har dubbelt så mycket (eller nästan) utrymme på skivan. Sedan är det en smaksak om man tycker det är bra eller inte.

Det lär finnas skivspelare med två lampmekanismer, där man kan vända skivan utan ytterligare åthävor, men någon standardvara är det inte. Jag undrar varför.

Sven Wickberg

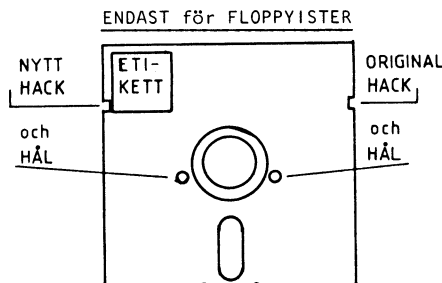
Har man grammofoon med dubbelsidig avläsning behövs inget extra hål. (red).

Jag har hört om och även sett s n y g g a

hålpptagningar. Man gör en noggrann mall, lägger in ett skydd för skivan och använder en hjälpuns vilken man snurrar mellan fingrarna tills det blir hål. Lika förfarande på motsatta sidan, men glöm inte att vända mallen rätt. Man skär med en vass kniv hacket i skivan, var noggrann!

Jag har fått låna disketter med hål men inte utan vidare kunnat läsa dem. På FD4 UD måste man sätta in en pappbit på motsatta sidan, det dög inte med tejp för skrivskydd, det uppfattades tydligen som felinsatt diskett och driven - som startar snurra när man sätter in disketten och stänger luckan - stoppar.

(red).



TILLBEHÖR!!

till helt oslagbara priser - jämför själv

VERBATIM Disketter:

minidiskett 525-01

25,- /st **220,-** /plastbox 10st

MOORE Datalistor:

För printning på 1/6". Stående A4, blått tryck, hålrat vänster kant.

55,- /1000 **150,-** /3000

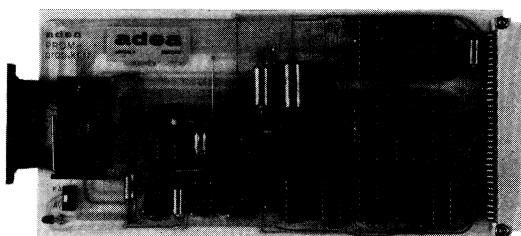
moms tillkommer på samtliga priser.

**...och mycket mer finns i vårt
INFORMATIONSBLAG
som du får GRATIS.**

DATA LAB

Box 2127 183 02 TÄBY
08-758 84 94

ABC 80 EPROM-PROGRAMMERARE



- Klarar de flesta 5V EPROM
- Passar ABC80/4680 bussen
- Utförlig bruksanvisning på svenska
- Lagervara
- Olika tillbehörsprogram finns

*** NYHETER ***

- Tillsatsmodul för 8741, 8748, 8749
- 8748 assembler
- Tillsatsmodul för Harris bipolära PROM
- PROM/RAM-kort för BYTE-WYDE 32k

adea Elektronik AB
Box 16015, 750 16 UPPSALA
Tel 018/ 10 06 02

Are You HELPless???

Då är Din räddning HELP,
ett nytt smart
programmeringshjälpmedel för

ABC80, ABC800 och FACIT DTC
för både M och C versionerna!

HELP ger din dator en ansiktslyftning!
Ur mängden nya kommandon kan nämnas:

FIND-söka text, VAR-variabellistning,
CAT-skinnehåll med filstorlekar,
Talomvandling, MEM-värdefulla
systemdata, m.m., m.m.!!!

Självklart har HELP också en smart
BILDSKÄRMSEDITOR så att Du kan
rädda felskrivna rader eller göra layouter
för menyer o. dyl.!

Kontakta oss så får Du ytterligare
information. Du kommer inte att förstå
hur Du klarat Dig utan HELP förut!

X-DATA HB

Baldersvägen 22, 198 00 BÄLSTA
Tel. 0171-513 70

HELP är en produkt från X-DATA HB och H-DATA HB.

FLEXSKIVOR

Rätt flexskiva till rätt disk. Billigast hos
importören. Låter sunt eller hur?

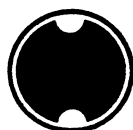
OMNI kvalitets-flexskiva 5 1/4" får du från oss
till priser som knappast någon kan konkurrera med.

Single side/Single density	10 st.	230:-
Single side/Double density	10 st.	240:-
Double side/Double density	10 st.	340:-
Flip Floppy/Single side	10 st.	340:-

Flip/Floppy är värdbar och kan använda båda sidorna i en Single side disk. Den lagrar alltså dubbelt så mycket information som en vanlig skiva.

Priset inkluderar MOMS, FRAKT, POSTFÖRSK.
samt BIBLIOTEKS/FORVARINGSBOX i hårdplast

RIKO DATA verkar också som en länk mellan svenska och finska programmerare och användare. Kommande billiga och avancerade program: Prov-analys, statistik, regressions-, faktoranalys samt ordbehandling för kassett/skiva. Vi söker program för utbildning.



RIKO DATA

Box 2082
230 41 BARA
Tel. 040-440737

smartaid[®] III

- * Automatisk uppstart av SMARTAID-3 funktioner och printerrutiner
- * Bildskärms"dump", skärmens innehåll skrivs ut på skrivare
- * Bildskärmseditorn förbättrad med bl.a. delete- och insert-funktion
- * Stoppfunktionen vid programkörning, som möjliggör körning av programmen en bit i taget med eller utan TRACE-funktionen
- * Multianpassad till såväl äldre som nyare ABC80 samt även för 80 tecken
- * Egen printerrutin gör det möjligt att använda den och ABC80 tillsammans med skrivare utan att behöva ladda in printerrutin i arbetsminnet eller ansluta till en flexskiveenhet.

HELP	skriver ut alla SMARTAID-3's och ABC80's kommandon på skärmen
REN	omnumrerar även <u>delar</u> av programmet.
LIST	listar programmet utan att de första rullar förbi. Med pilarna kan scrollning ske framåt och bakåt. Listning av valda programdelar kan ske till fil t.ex. skrivare.
Ü	I stället för LIST kan du använda en förkortning, bokstaven ü
CHANGE	byter ut en gammal variabel till en ny i hela programmet.
FIND	söker och skriver ut hela de programrader, där sökt begrepp finnes.
OLD	tar fram föregående program efter NEW eller RESET.
PEEK	listar minnesinnehåll i decimal-, hexadecimal- och ASCII-form
LIB	visar ett bibliotek på flexskiva (DIR för 8 tums skivor)
SYS	visar kvarvarande- och program-minnesstorlek, olika pekare, systemtid i tim, min, sek m.m.
RESUME	återupptar programkörningen efter CTRL C eller STOP med bibehållna variabelvärden.

FOR...NEXT FOR..NEXT loop kan användas som direkt kommando (utan program)

Dessutom finns AUTO, DEL VAR och START (se SMARTAID II).

NY DESIGN SMARTAID-3 levereras i en robust låda med svart stålplåt och minimalt djup bakom tangentbordet, mindre än 20 mm.

BESTÄLL DIN SMARTAID-3 IDAG !

För demonstration eller beställning kontakta närmaste återförsäljare eller ring OWOCO

OWOCO AB

Kvarnbergsvägen 25, 141 45 HUDDINGE Tel. 08/774 02 90